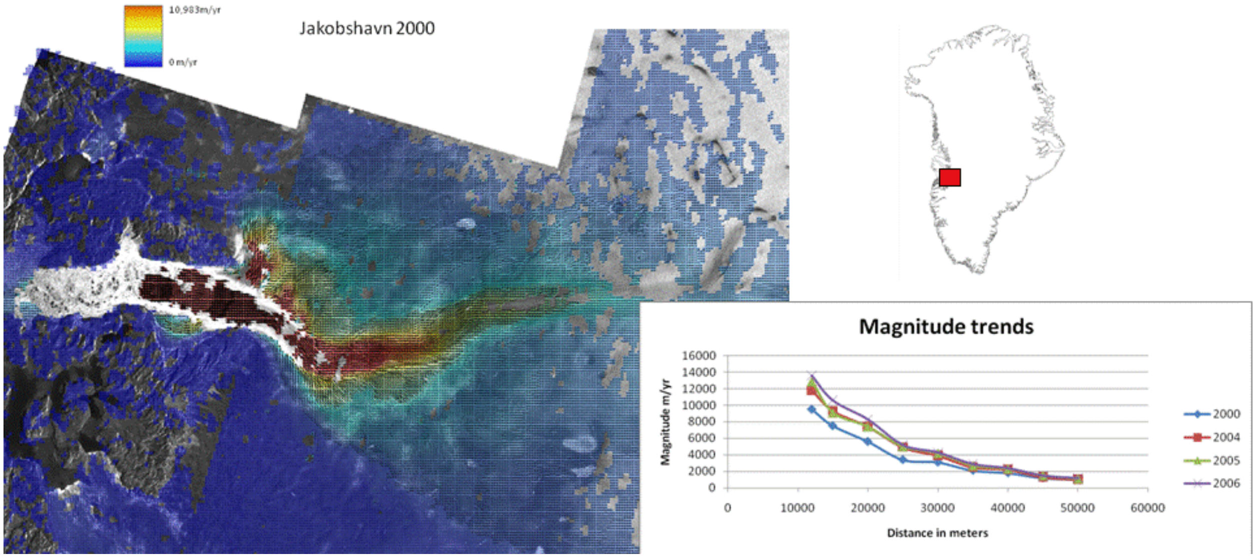
Velocity Trends for Jakobshavn Glacier, Greenland for the Years 2000, 2004, 2005, and 2006 Including Procedure Manuals



BPRC Technical Report Number 2008-01

Byrd Polar Research Center
The Ohio State University
Columbus Ohio 43210-1002

Compiled in 2008 by the

BYRD POLAR RESEARCH CENTER

This report may be cited as:

Farness, Katy and Kenneth Jezek. 2008. *Velocity Trends for Jakobshavn Glacier, Greenland for the Years 2000, 2004, 2005, and 2006 including Procedure Manuals.* BPRC Technical Report No. 2008-01, Byrd Polar Research Center, The Ohio State University, Columbus, Ohio, 60 pages.

The Byrd Polar Research Center Report Series is edited by Lynn Tipton-Everett.

Copies of this and other publications of the Byrd Polar Research Center are available from:

Publication Distribution Program
Byrd Polar Research Center
The Ohio State University
1090 Carmack Road
Columbus, Ohio 43210-1002
Telephone: 614-292-6715

Velocity Trends for Jakobshavn Glacier, Greenland for the Years 2000, 2004, 2005, and 2006 Including Procedure Manuals

Katy Farness and Kenneth Jezek

December 9, 2008

Byrd Polar Research Center
The Ohio State University
Columbus, Ohio

BPRC Technical Report 2008-01

Acknowledgements

This work was supported by the Center for Remote Sensing of Ice Sheets (CReSIS), from a grant sponsored by the National Science Foundation. OSU received their funding as a sub-award (Award #: 22000219: FY2005-109C) from the University of Kansas. A special thanks to James Miller and Xiaoqing Wu from the Vexcel Corporation for their never wavering support to my many and unending stream of questions.

Table of Contents

1.0 Introduction 1

2.0 Data Sets 2

3.0 Processing Plan 2

4.0 Methodology 3

4.0.1 InSAR Methodology 3

4.0.2 Optimization of Interferometric Parameters used in RAMS VphaseTool 5

4.0.3 Procedure for Creating Final Filtered Velocities from Multiple RAMS Velocity Products 9

5.0 Observations 12

List of Figures

Figure 1. Location of Jakobshavn Glacier 1

Figure 2. Processing flow chart 3

Figure 3. Registration tie points on stable rocky outcrops and within glacier flow regime 4

Figure 4. General areas that were subsetted so that different parameters could be applied to try and obtain optimal velocities. 8

Figure 5. Examples from the 2006 dataset of erroneous velocities both of bad magnitudes and bad directions 10

Figure 6. Example of velocity control point distribution over rocky outcrop using the RAMS VCP edit tool 11

Figure 7. Profile line used to plot velocities 12

Figure 8. Velocity coverage map for the year 2000 13

Figure 9. Velocity coverage map for the year 2004 14

Figure 10. Velocity coverage map for the year 2005 15

Figure 11. Velocity coverage map for the year 2006 16

Figure 12. 2006 velocity coverage map depicting areas of sparse velocity coverage within the glacier main drainage 17

Figure 13. Velocity profile: 2000 10,996m/yr. 17

Figure 14. Velocity profile: 2004 12,505m/yr. 18

Figure 15. Velocity profile: 2005 13,341m/yr 18

Figure 16. Velocity profile: 2006 13,771m/yr. 19

Figure 17. Magnitude trends plotted at specific distance intervals. 19

List of Tables

Table 1. Available satellite data sets 2

Table 2. Chip size parameter values relative to feature size in meters 6

Table 3. Search radius parameter values relative to velocity in meters per year 7

Table 4. Optimal search radius and chip size parameters used for year 2000 velocity calculations 9

Table 5. Optimal search radius and chip size parameters used for years 2004, 05, and 06 velocity calculations 9

Table 6. Baselines and bandwidths for interior swath pairs. 12

Addendums

Addendum I. Procedure for creating final filtered velocities from multiple RAMS velocity products 20

Addendum II. Procedure for creating velocity control points to propagate from one overlapping frame pair to the next 27

Addendum III. RAMS Phase I Processing 29

Addendum IV. RAMS Phase II Processing 42

1.0 Introduction

This report describes procedures used to process RADARSAT synthetic aperture radar data acquired over Jakobshavn Glacier, Greenland (Figure 1). Interferometric pairs are used to compute ice sheet surface velocities for the years 2000, 2004, 2005, and 2006 during the approximate months of November through early January of the following year. The methodology and results discussed in this report are provided in partial fulfillment of OSU's agreement to provide remote sensing data to the Center for Remote Sensing of Ice Sheets (CReSIS) science requirements. CReSIS's vision is to advance understanding of polar ice sheets sufficiently for the development of models that reliably predict future ice sheet contributions to sea level rise under prescribed changes in climate.

Jakobshavn Glacier, Greenland's largest glacier, drains approximately 6.5% of the ice sheet and is a key factor in the mass balance of the Greenland Ice Sheet. Although the interior of the ice sheet is in a state of balance, the coastal region has continued to thin (Krabill et. al, 2000). While there has been variability in the velocity of Jakobshavn over time the ice front has steadily retreated (Figure 1). It wasn't until the year 2000 that an increase in ice flow velocities was recorded and an increasing trend observed.

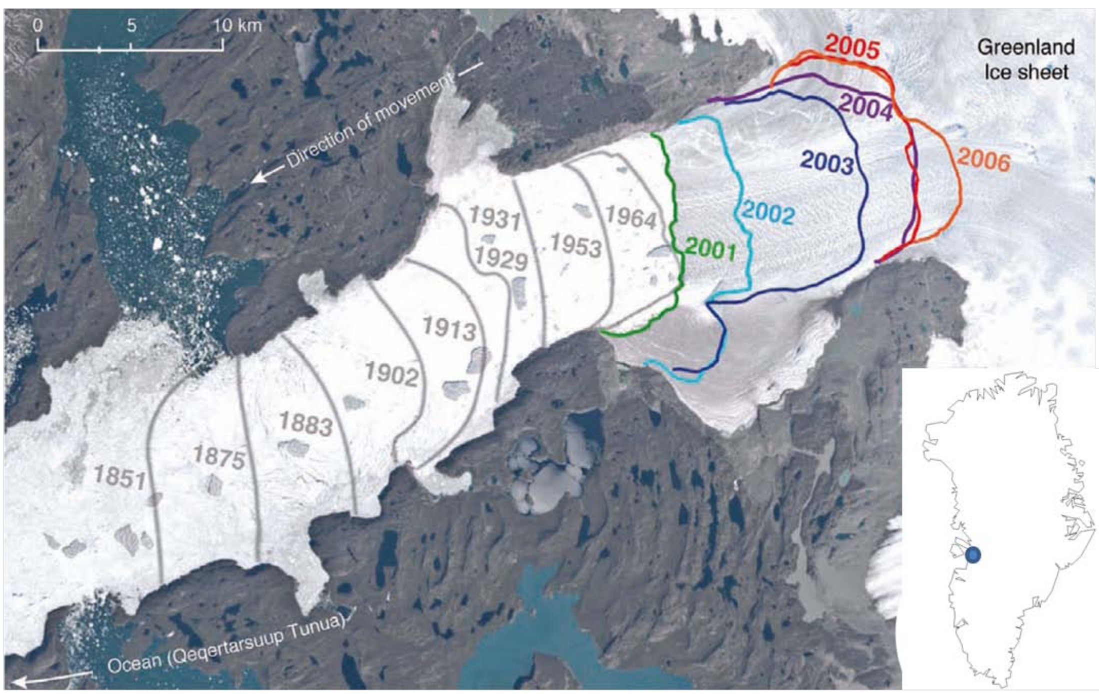


Figure 1. Location of Jakobshavn Glacier (lower right corner). Illustration showing ice front retreat since 1851 overlain on 2001 Landsat image. (Sources: NASA/Goddard Space Flight Center Scientific Visualization Studio. Historic calving front locations courtesy of Anker Weidick and Ole Bennike).

2.0 Data Sets

To calculate velocities for the Jakobshavn Glacier we used 24-day repeat pass RADARSAT-1 fine beam descending orbit pairs. Coverage consisted of multiple adjacent overlapping orbit pairs per year. See Table 1 for the breakdown on the available satellite orbit pairs used for this study.

Table 1. Available satellite data sets.

Year	Orbit	Acquisition date	24-day repeat orbit	Acquisition date
2000	26123	November 5	26466	November 29
	26566	December 6	26909	December 30
	26666	December 13	27009	January 6, 2001
2004	47389	December 2	47732	December 26
	47146	November 15	47489	December 9
	47589	December 16	47932	January 9, 2005
2005	52534	November 27	52877	December 21
	52634	December 4	52977	December 28
	52734	December 11	53077	January 4, 2006
2006	57679	November 22	58022	December 16
	57779	November 29	58122	December 23
	57879	December 6	58222	December 30

In addition to the satellite data, a digital elevation model (DEM) was used to terrain correct the image data and to remove the topography-related phase for projecting velocity measurements onto the surface plane of the ice sheet. The DEM was provided by B. Csatho who merged ICEsat and shape-from-shading data. Because InSAR calculates relative displacements, an adjustment of absolute motion for several points in each frame pair needs to be applied. These velocity control points (VCPs) were placed on stable rocky outcrops for coastal frame pairs. Velocity control was propagated onto the interior ice sheet by successively correlating points from one adjacent frame pair to the next as they approached the more featureless interior (see addendum II).

3.0 Processing Plan

Data processing for the Greenland mission was structured from the procedures and infrastructure established during the RADARSAT Antarctic Mapping Project (RAMP) (Jezek, 2008). In order to capture the ever increasing velocity from the interior basin to the glacier terminus, a modified approach was developed (Figure 2). A detailed procedure manual can be found in Addendum I. An account of the methodology follows in the next section.

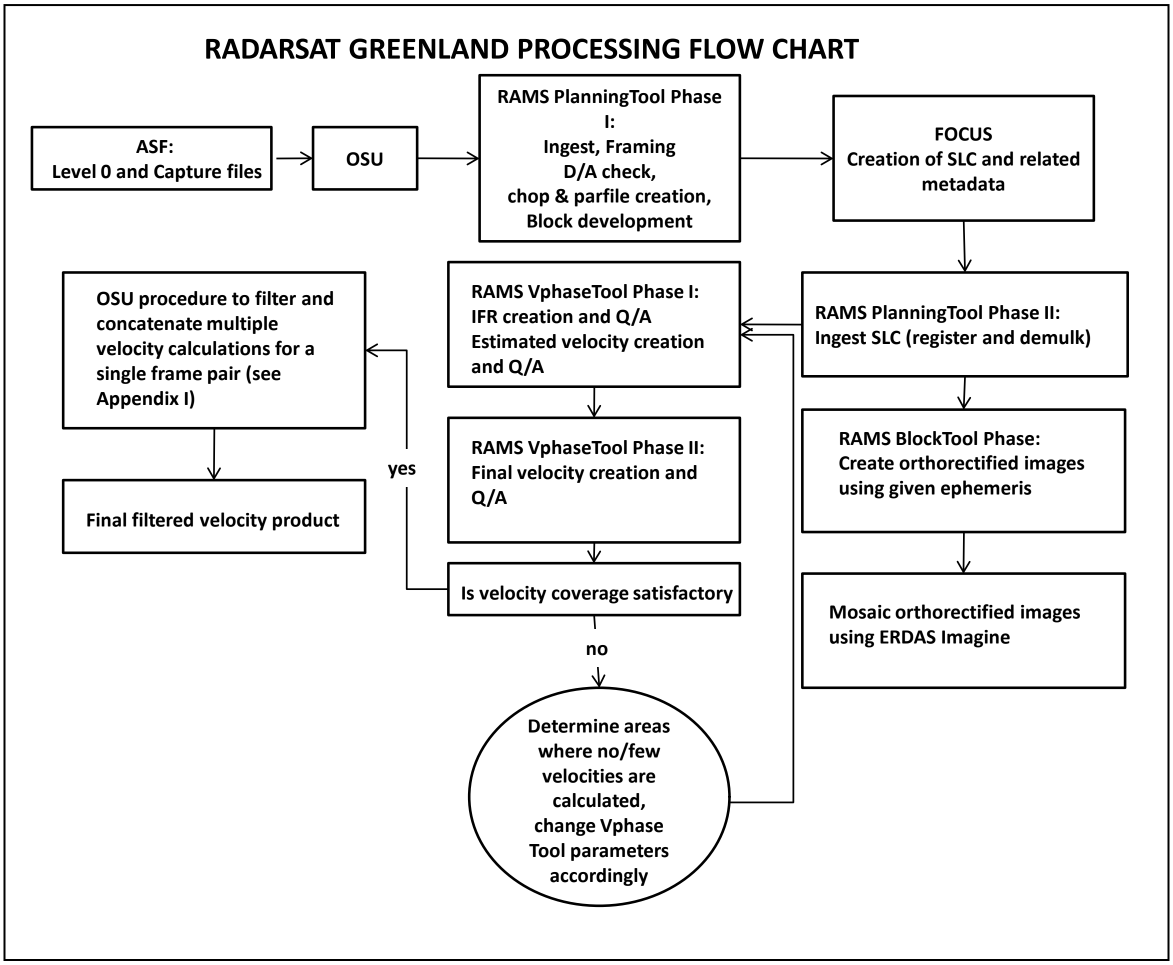


Figure 2. Processing flow chart.

4.0 Methodology

This section gives an overview of RAMS (RADARSAT Antarctic Mapping System) Interferometric (InSAR) processing taken from Jezek, 2008. For more detailed information please refer to the RAMP project document available in the Knowledge Bank at website http://hdl.handle.net/1811/33986. In addition, an explanation of the VphaseTool parameters (also referred to as correlation chips) is given and how they were optimized in order to capture the diversity in ice flow velocity for any selected frame pair.

4.0.1 InSAR Methodology

Image registration begins simply by registering the data using the published ephemeris. The next refinement proceeds by selecting a small number of tie points and computing an affine transformation to roughly align the images. A hierarchy of correlation methods is subsequently applied to compute the precise registration and offset maps. The approach first tries to use the interferogram spectrum method because it yields the most accurate registration results. If the spectral method fails, the well known amplitude correlation method, which only uses the magnitude of the SLC data, is used. The correlation peak is computed by fitting a quadratic function to the correlation data. The computed offset of the correlation peak can be used to obtain either a speckle retracking or feature retracking result depending on the size of the correlation chips.

Tie points are automatically generated, followed by manual inspection. All tie points generated in sea ice are manually deleted. Tie points with bad or questionable registration between the two frame pairs are deleted. Tie points located both on stable features like rocky outcrops and within the glacier flow regime are examined and edited so that like-features are matched (Figure 2).

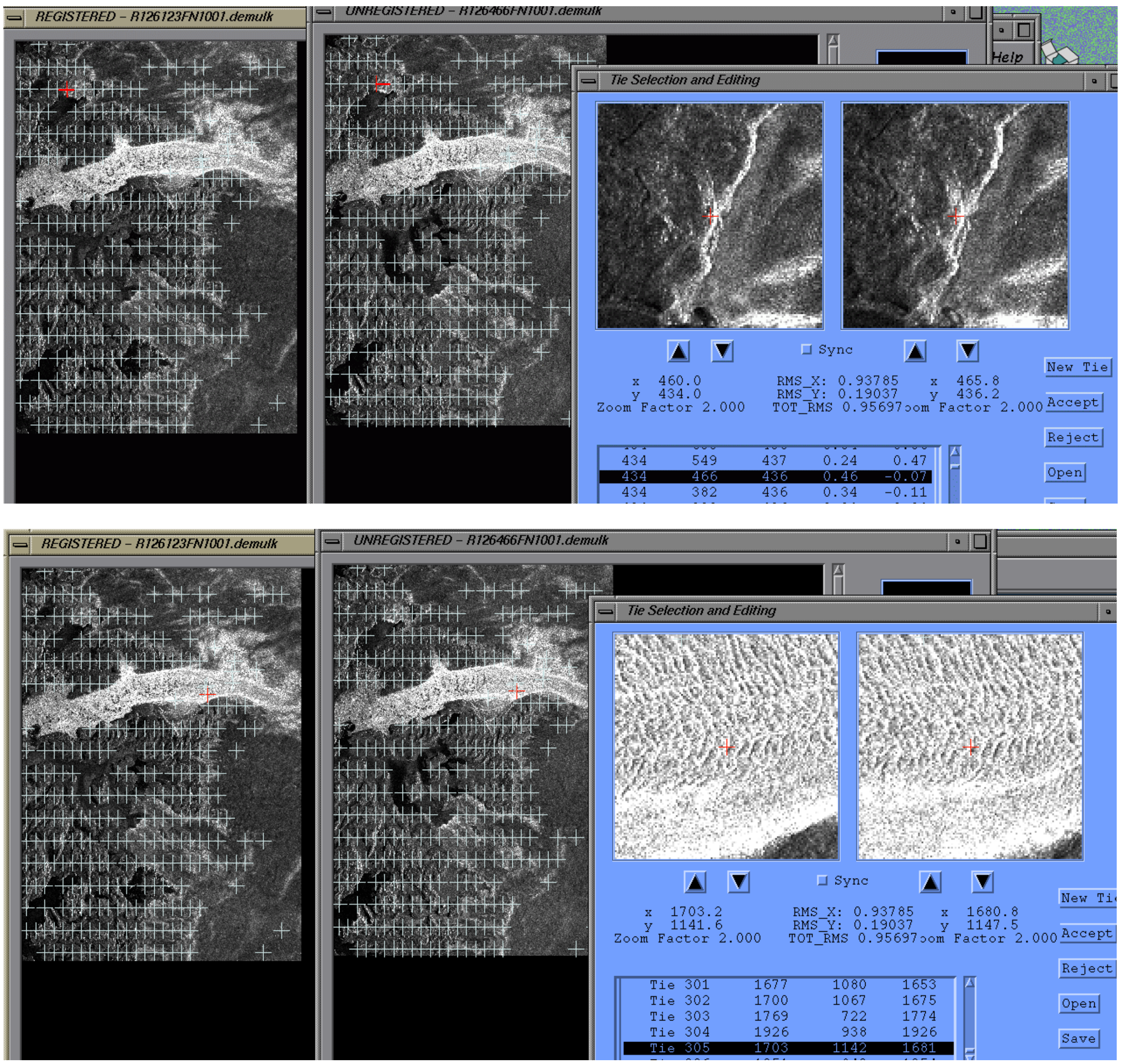


Figure 3. Registration tie points on stable rocky outcrops (top) and within glacier flow regime (bottom).

The interferogram is filtered using the Goldstein α filter. Two algorithms are then used for phase unwrapping, these are the miniMax method and a combination of miniMax and the minimum cost flow (MCF) method. The miniMax method utilizes the regional clustering property of residues and groups residues into different clusters with zero net residues under the criterion of minimizing the maximal cut length. It is optimal in the sense that it minimizes the maximum length of cut lines. The MCF is optimal in the sense that it minimizes the total length (or weighted length) of cut lines. It is however computationally inefficient. The perfect combination of miniMax method and the MCF method minimizes the maximum cut length and also minimizes the total cut length for each cluster and thus provides a good compromise between quality and efficiency. The interferograms are then corrected for the local elevation using the DEM.

Two velocity map products can be created from a single interferogram. In one case, the azimuth offset and the unwrapped phase are used to derive the surface movement assuming the ice is moving on the surface. In the other, the same azimuth offset is used but the range offset is used instead of the unwrapped phase. Since we have only descending swath pairs (where as in RAMP we had both ascending and descending) the final velocity computation utilizes only the following combinations:

Azimuth offset, range phase
Azimuth offset, range offset
Descending range phase
Descending azimuth offsets

Generally the descending azimuth offsets were not used as these are the poorest quality estimates. Additionally, feature retracking can be used for the faster parts of the glacier. Feature retracking relies on much larger reference image chip sizes than would normally be used for phase or speckle matching methods.

4.0.2 Optimization of Interferometric Parameters used in RAMS VphaseTool

In order to capture the diverse velocity field of the Jakobshavn Glacier Basin multiple sets of RAMS Vphase parameters (Table 2) had to be used. A detailed description of all the RAMS VphaseTool Phase I parameters (registration and creation of interferograms) was provided by the Vexcel Corporation, the developers of RAMS, and is available in addendum III. This section will focus only on the parameters that were optimized to obtain better velocity coverage.

In the creation of the interferograms, the three prominent parameters used for estimating a range of velocities are the 1) lower coherence threshold, 2) offset chip size and 3) offset search radius, all other default parameters were used.

The lower coherence threshold is the coherence value below which tie points of the affine transform and offset map are not use. Coherence values range from 0 to 1 (bad to good). The more dynamic the change over a period of time (for this study, 24 days) the poorer the coherence. Given the dynamic nature of the Jakobshavn Glacier and its surroundings, coherence is often poor. For this reason the coherence threshold is set very low at 0.001 for all calculations. The down side is that a low threshold will not filter out bad tie points. Bad tie points will, in effect, give bad velocity vectors. To compensate for this, we developed a procedure for filtering out such erroneous vectors outside of RAMS discussed in section 4.0.3 (see addendum I, step 4).

Generally low coherence meant that most of the analysis relied on feature retracking. Consequently, the chip size parameters are dependent upon the feature size or wavelength. Chip size can be determined by measuring the feature size (usually the larger features) in your area of interest, taking into consideration the feature orientation with respect to azimuth and range ((x, y) or pixel and line). Note that for the fine beam single look complex (SLC) data that we used for creating interferograms, the pixel size is approximately 5m in azimuth and 8m in rang. Admissible values are integer powers of 2. Table 2 relates chip size parameter values to feature size in meters per year.

Table 2. Chip size parameter values relative to feature size in meters.

Integer power of 2	Value	Feature size: azimuth (value x 5m)	Feature size: range (value x 8m)
2^2	4	20m	32m
2^3	8	40m	64m
2^4	16	80m	128m
2^5	32	160m	256m
2^6	64	320m	512m
2^7	128	640m	1024m
2^8	256	1280m	2048m

The search radius parameters are dependent upon the displacement of features over time. Obtain in situ measurements (Vm/a) from published data if available, else measure feature displacement (Xm) using the 2 - 24 day repeat pass 25m orthorectified images created in the BlockTool Phase (see table 1) taking into consideration the flow direction with respect to azimuth and range. Admissible values are any number greater than 0. The basic equations for calculating the pixel and line value for the search radius parameters is as follows, where eq. 1 is for known annual velocities and eq. 2 is 24-day measured displacements.

$$[(Vm/a \div 365d/a) * 24d] \div 5m/pix \quad \text{(or 8m/pix for range pixel size)} \qquad \text{Eq 1}$$

$$Xm \div 5m \text{ (or 8m for range pixel size)} \qquad \text{Eq 2}$$

Thus the faster the ice flow, the larger the search radius needs to be in order to capture the feature offsets. That said, the larger the search radius the more global the search results where as the smaller the search radius the more localized the search results. The global approach will also introduce velocity errors due to the larger averaged area. Such errors need to be manually edited out as outlined in addendum I. Table 3 relates velocity in meters per year to search radius parameters using equation 1.

Table 3. Search radius parameter values relative to velocity in meters per year. Search radius values are rounded to the nearest integer.

Velocity m/yr relative to azimuth and range (x,y)	Azimuth (pixel) search radius	Range (line) search radius
100m/yr	2	1
500m/yr	7	4
1000m/yr	13	8
2500m/yr	33	20
5000m/yr	66	41
7500m/yr	99	62
10,000m/yr	132	82
12,500m/yr	164	103
15,000m/yr	197	123
17,500m/yr	231	144
20,000m/yr	263	164

Figure 4 illustrates the general areas that were subsetted and where different parameters were applied to try and obtain optimal velocities. Due to the relatively slower velocities and more coherent features witnessed in 2000, only four general areas were targeted for subsetting. Likewise, due to the faster velocities and with the more chaotic nature of the crevasse features, five general areas were targeted for subsetting for the years 2004, 05, and 06.

The first step was to process full frame pairs using a combination of complex and feature tracking methods. Due to some limitation of the RAMS software the maximum search radius allowed was 16x16 and the maximum chip size was 64x64. Since most of the crevasse features on Jakobshavn measure a few hundred meters, 64x64 chip size should have been adequate. However and after many tests, it was found that a 128x128 chip size worked better. In a few tests even larger chip sizes were attempted (up to 256x256) but in most cases the results were not much improved and in some cases a little noisier. In addition, the larger chip sizes took much longer processing time, therefore a chip size of 128x128 was deemed optimal.

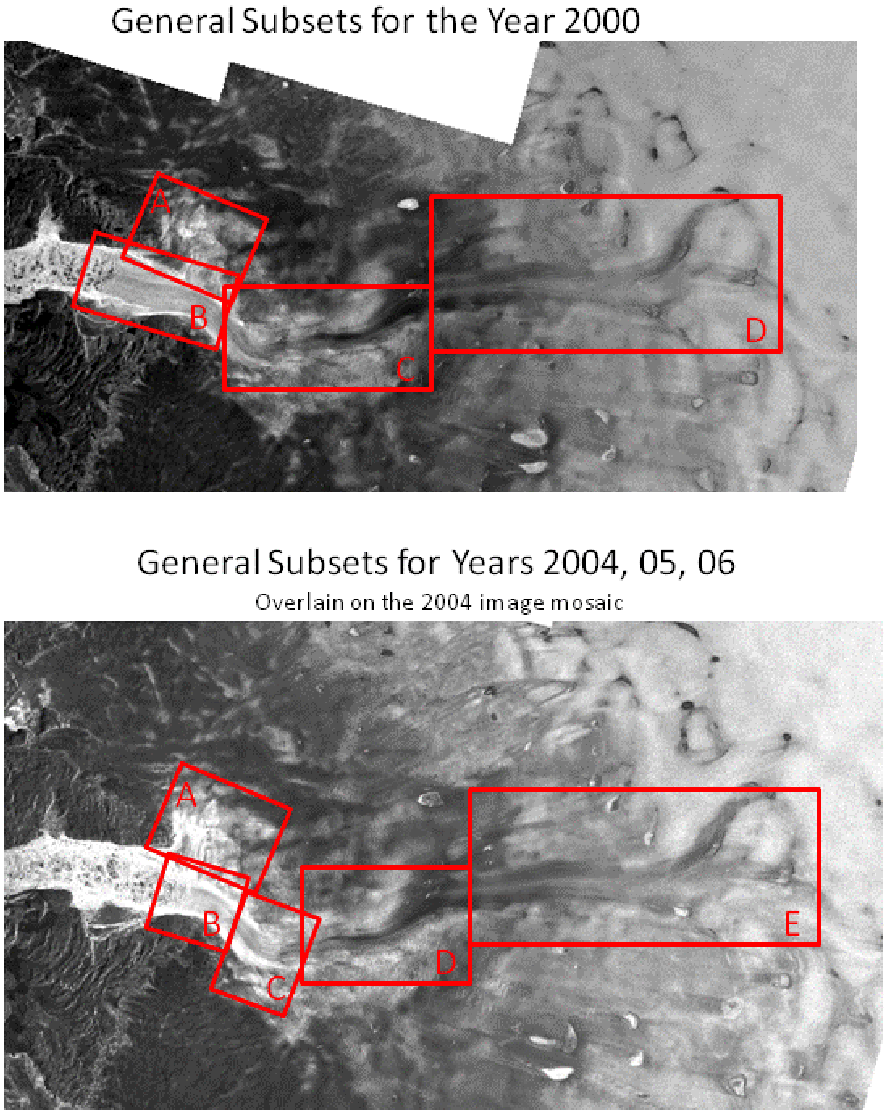


Figure 4. General areas that were subsetted so that different parameters could be applied to try and obtain optimal velocities.

As for the search radius, the 16x16 that we initially used was not sufficient to track the larger feature displacements downstream toward the glacier terminus (see table 3). Keep in mind though that the larger the search radius the more global the averaging becomes and thus has a tendency to introduce more error. Parameters that worked well for the year 2000 subsets are listed in Table 4.

Table 4. Optimal search radius and chip size parameters used for year 2000 velocity calculations.

Area	Search Radius	Chip Size
Subset A	60x60 80x80	128x128
Subset B	128x128	128x128
Subset C	80x80	128x128
Subset D	60x60	128x128

For the years 2004, 05, and 06, a larger search radius seemed warranted - especially towards the terminus of the glacier where the offsets were the largest. Unfortunately, for radii over 200 pixels (ie: 256x256) the results were extremely noisy and extremely difficult to quality control. So these were not used in creating the final filtered velocity (see section 4.0.3). In the end the parameters in Table 5 provided the best 'usable' results.

Table 5. Optimal search radius and chip size parameters used for years 2004, 05, and 06 velocity calculations.

Area	Search Radius	Chip Size
Subset A	60x60 80x80	128x128
Subset B	150x150 192x192	128x128
Subset C	80x80 128x128 150x150	128x128
Subset D	80x80	128x128
Subset E	60x60	128x128

4.0.3 Procedure for Creating Final Filtered Velocities from Multiple RAMS Velocity Products

A detailed description of all the RAMS VphaseTool Phase II parameters was provided by the Vexcel Corporation, the developers of RAMS, and is available in addendum IV. This section will focus on the filtering and blending of multiple datasets and velocity control point selection and propagation.

Multiple versions (interferometric subsets) were created for each of the frame pairs so that the optimal chip size and search radius parameters could be applied based on feature wavelength and glacier speed (see Tables 2 and 3) and turned into RAMS velocity products. The subsets all have unique start pixel and line numbers as well as the number of pixels and lines depending on the area of interest. Since we had so many velocity products (subsets), a methodology was created to blend all these datasets together. A detailed description of the procedure for creating final filtered velocity from multiple RAMS velocity products can be found in addendum I. Before we could blend the subsets together we needed to remove any offending data point.

Basically this entails extracting the magnitude and X and Y component of velocity from the RAMS velocity product. Using ERDAS Imagine we then added the map projection and converted the raster files to an ASCII file containing the map coordinates (polar stereographic projection), magnitude, Vx, and Vy. From there we calculated the flow direction (angle) in an EXCEL spreadsheet [degrees(ATAN2(vx, vy))]. The ASCII file was converted into a shape file with attribute table using ARC/Info. This was done for each individual subset. From there we needed to remove any offending data point. An offending data point is defined as any vector pointing contrary to the known slope direction or contrary to observable flow features or for which magnitudes were obviously incorrect (Figure 5).

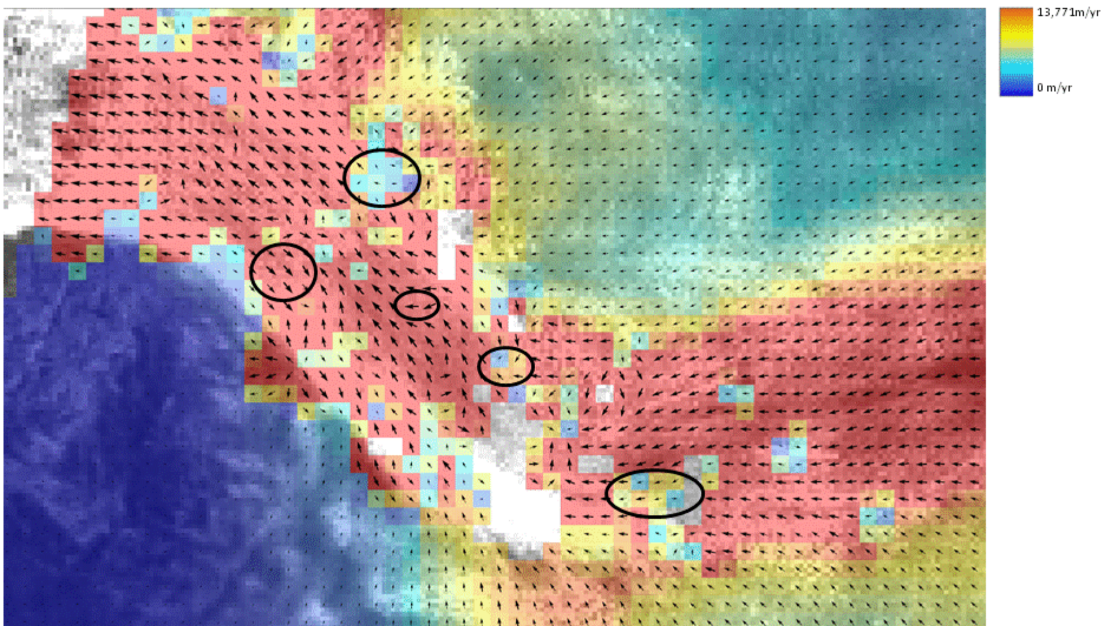


Figure 5. Examples from the 2006 dataset of erroneous velocities both of bad magnitudes and bad directions.

For Jakobshavn in particular, we defined erroneous flow direction as going from left to right across the frame or Vx < 0 (negative), where Vx is the x or azimuth component of velocity and removed them using a logical expression in ARC. Next an aggressive visual inspection was made on each individual subset and any velocity vector that appeared erroneous, whether in direction or magnitude, was manually deleted from the attributes file. This procedure was repeated until the operator was satisfied with the results. After editing each subset, we converted each shapefile back into magnitude, Vx, Vy raster files (ERDAS Imagine format). In ERDAS we then used the mosaic tool with an averaging scheme to concatenate all the subsets together into a final filtered velocity.

Since interferometry only gives relative displacements, an adjustment needs to be applied based on knowledge of absolute motion for several points in each image. This is done by using velocity control points. For frame pairs that contained a lot of rocky outcrops, we could easily add zero velocity control (Figure 6). However as the coverage moved in toward the interior where the topography was practically featureless, a methodology was needed to glean the

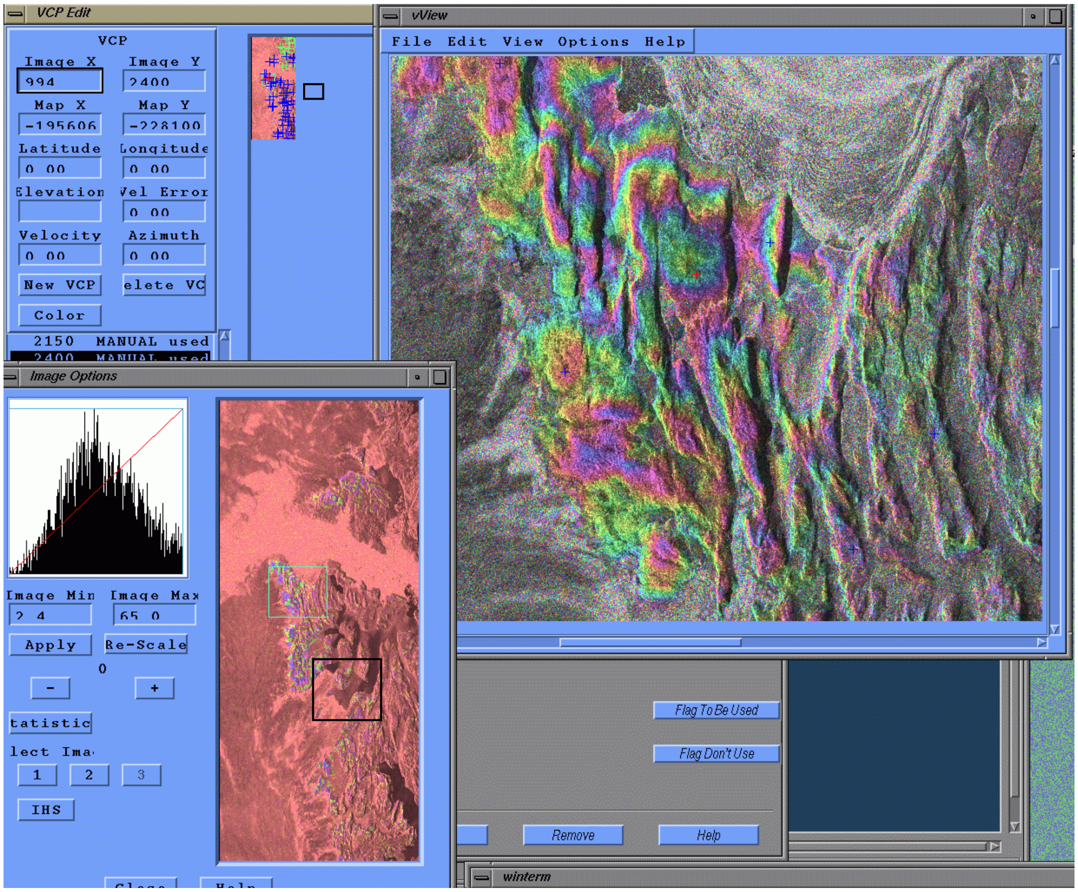


Figure 6. Example of velocity control point distribution over rocky outcrop using the RAMS VCP edit tool.

velocities calculated from the coastal frames and migrate those via the overlapping portion of adjacent swaths. A detailed description of velocity control point propagation is described in addendum II. The initial steps are the same as described in the preceding paragraph up to creation of the shape file with attribute table in ARC/Info. At this point we converted the map coordinates into latitude and longitude, which is the format needed to ingest back into RAMS. We exported the attributes file as a text file and edited this text file into a RAMS friendly format and ingested the information back into the mission as velocity control points.

5.0 Observations

To capture the progression of a feature downstream through time, a velocity profile was drawn along a flow line as determined by the velocity flow vectors and flow patterns in the SAR image (Figure 7). In addition, the glacial terminus was derived from the corresponding SAR image for each year and was used as a termination point for the profile line.

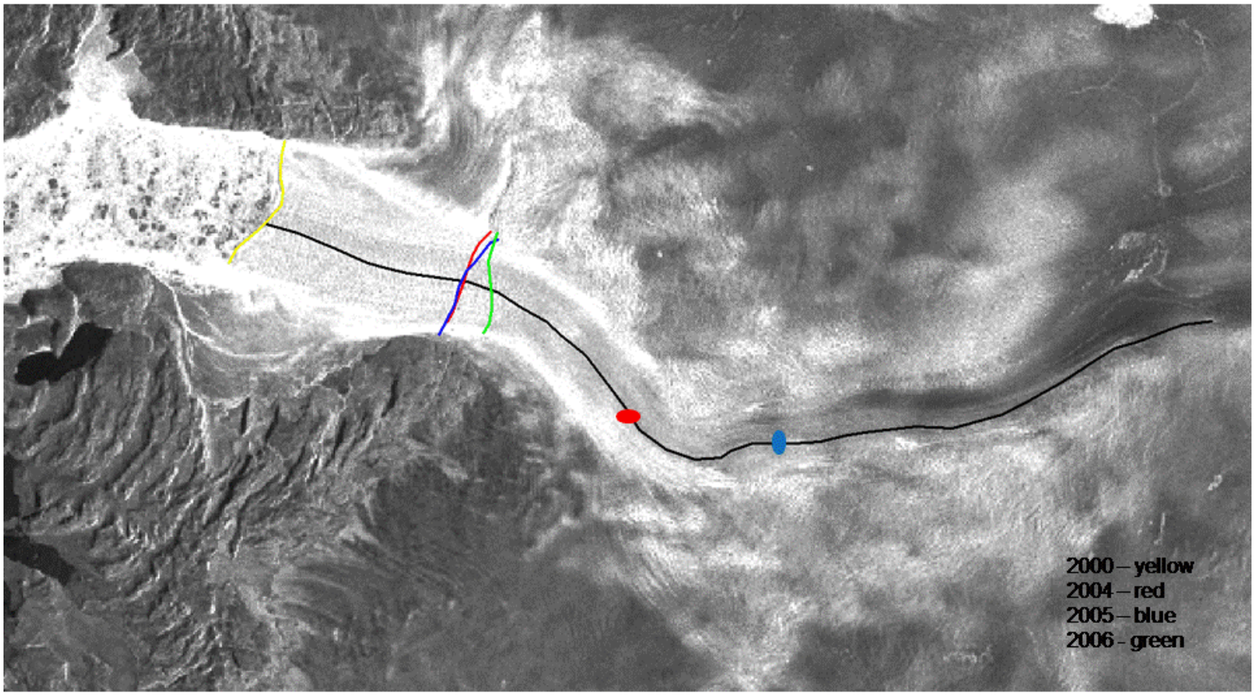


Figure 7. Profile line used to plot velocities. Estimated glacial terminus plotted as follows: 2000 yellow, 2004 red, 2005 blue, 2006 green. Red dot is approximately 20,000m mark (inward from the 2000 terminus). Blue dot is approximately 28,000m mark.

Velocity coverage maps were created in ArcInfo for each year (Figures 8-11). Note that the farther into the interior, especially for the years 2004 and 2006, overall coverage becomes very sparse.

We examined the refined baseline and Doppler bandwidth overlap for each of the interior swath pairs (Table 6) to see if that would explain the lack in coverage. Normally the smaller the baseline and larger the bandwidth the better the chances are for good registration between repeat pairs. Although the bandwidth was marginal for the 2006 pair, which could explain the poor velocity coverage, the baseline was also very low (good). Alternatively, the 2004 pair had a large baseline but a reasonable bandwidth. The large baseline could explain the poor velocity coverage, except the 2000 pair had and even larger baseline with a similar bandwidth and had reasonable results.

Table 6. Baselines and bandwidths for interior swath pairs.

Year	Orbit Pair	Baseline (refined)	Bandwidth
2000	26666-26909	917	934
2004	47589-47932	882	946
2005	52734-53077	138	1020
2006	57879-58222	42	636

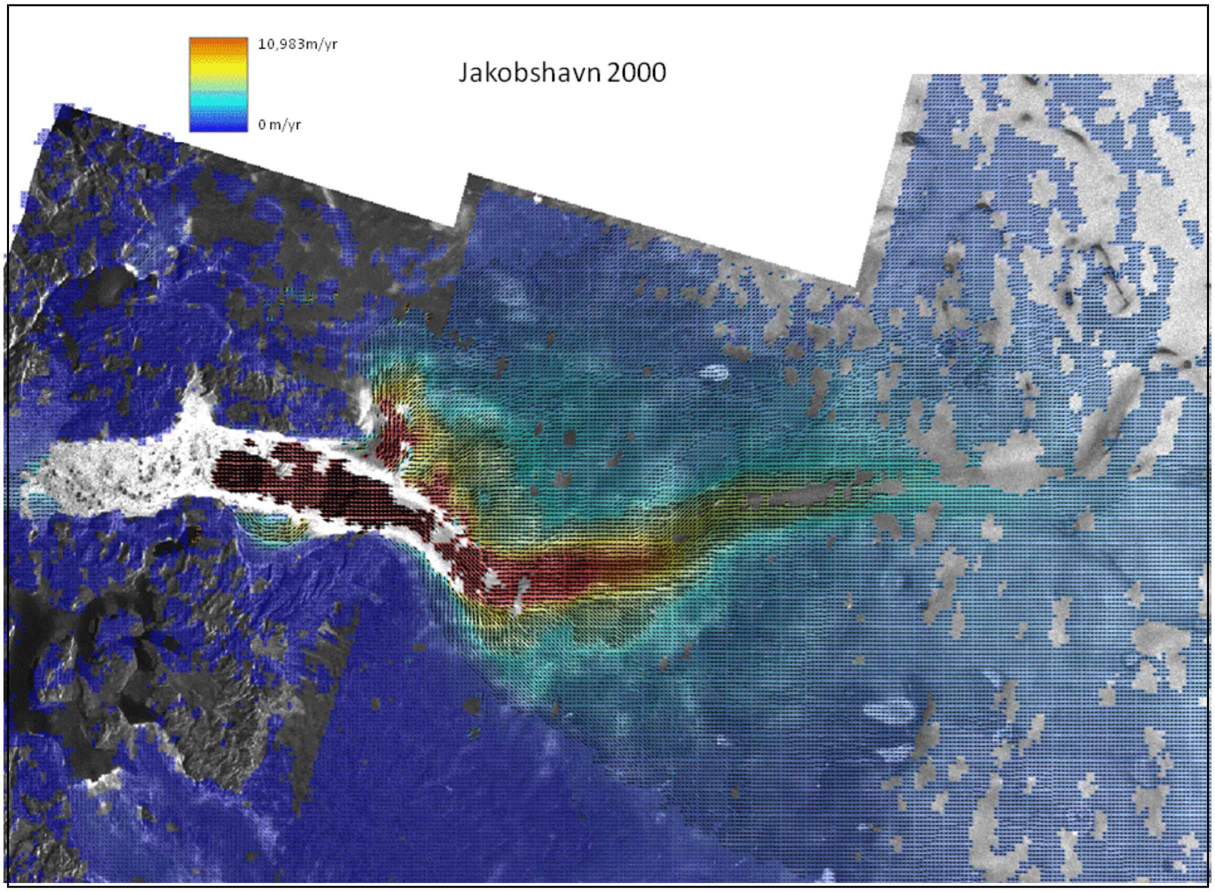


Figure 8. Velocity coverage map for the year 2000.

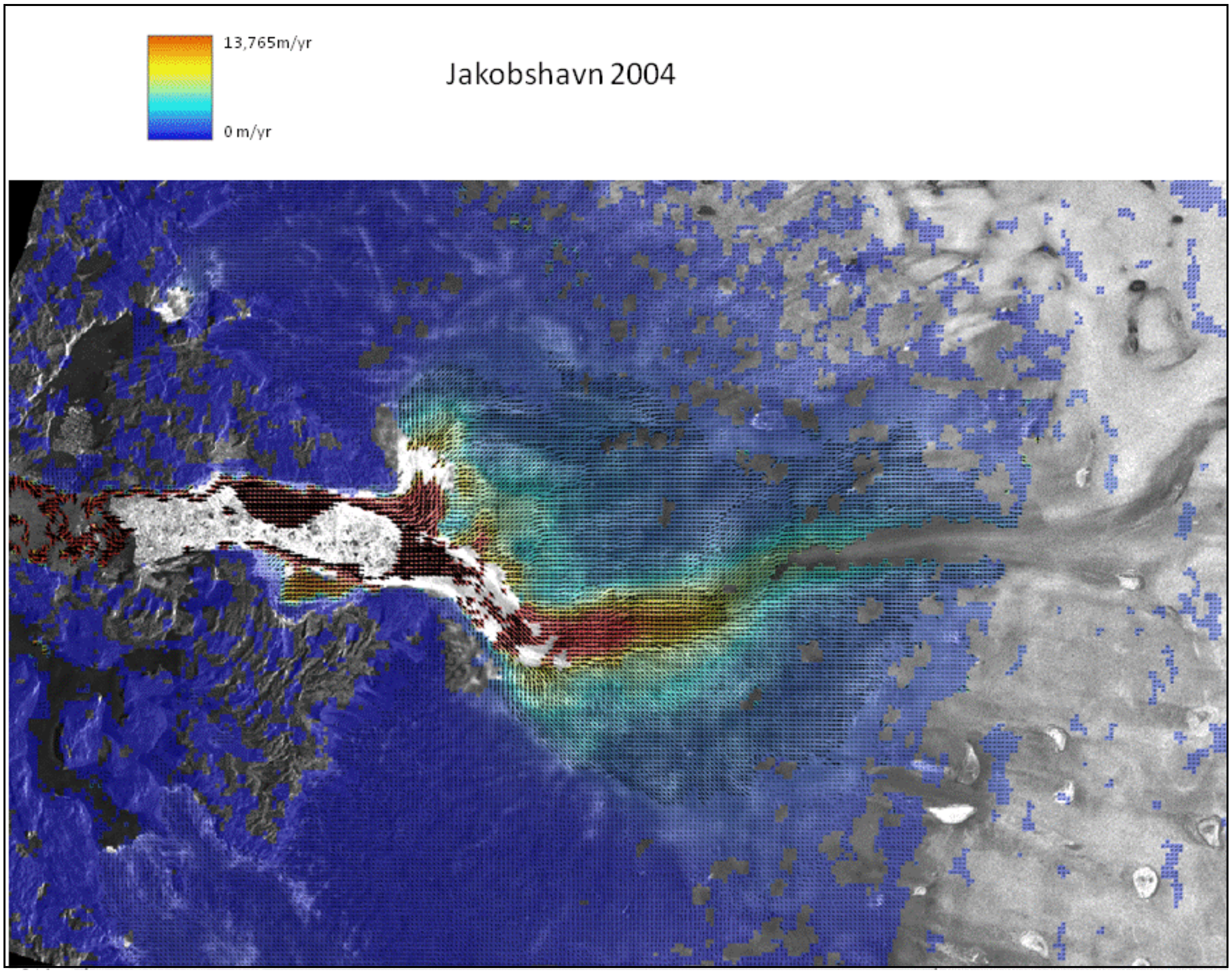


Figure 9. Velocity coverage map for the year 2004.

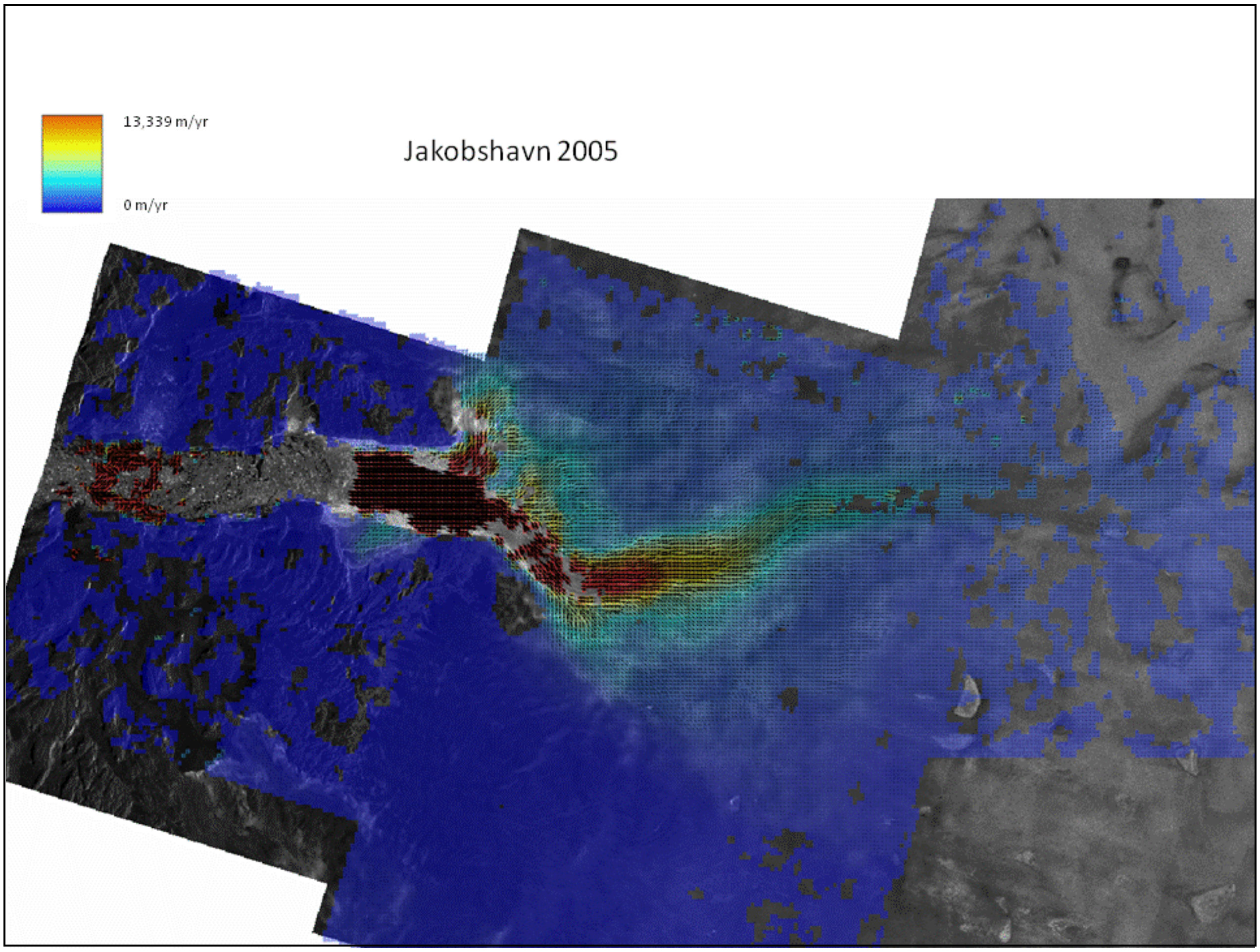


Figure 10. Velocity coverage map for the year 2005**.**

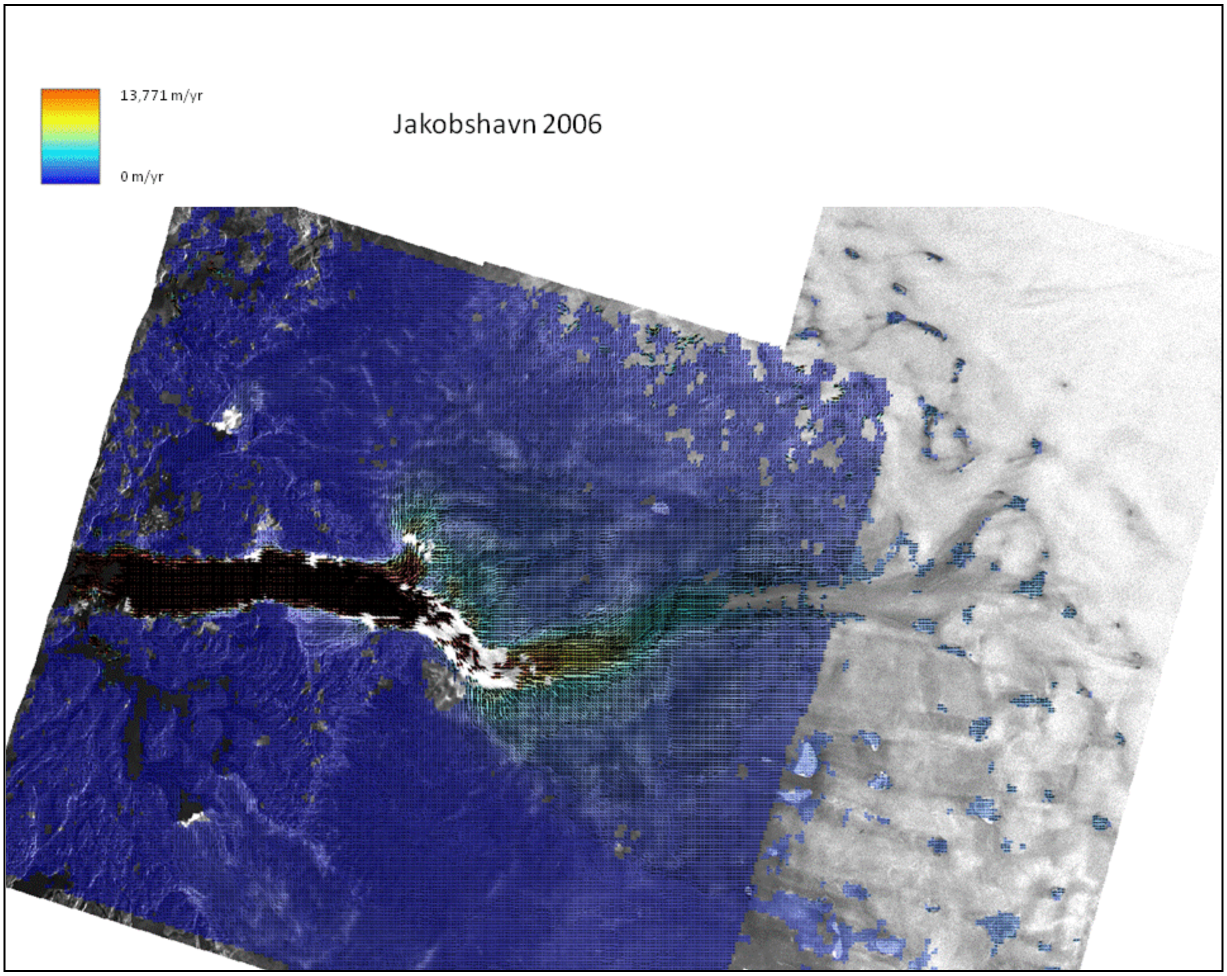


Figure 11. Velocity coverage map for the year 2006.

In all four cases there seems to be sparse coverage both in the upper flow regime and in the forward bend of the glacier (Figure 12). In both cases coherence was extremely low, but also tracking features were nearly impossible due to either feature deformation over time (B) or that the features were indistinguishable (A).

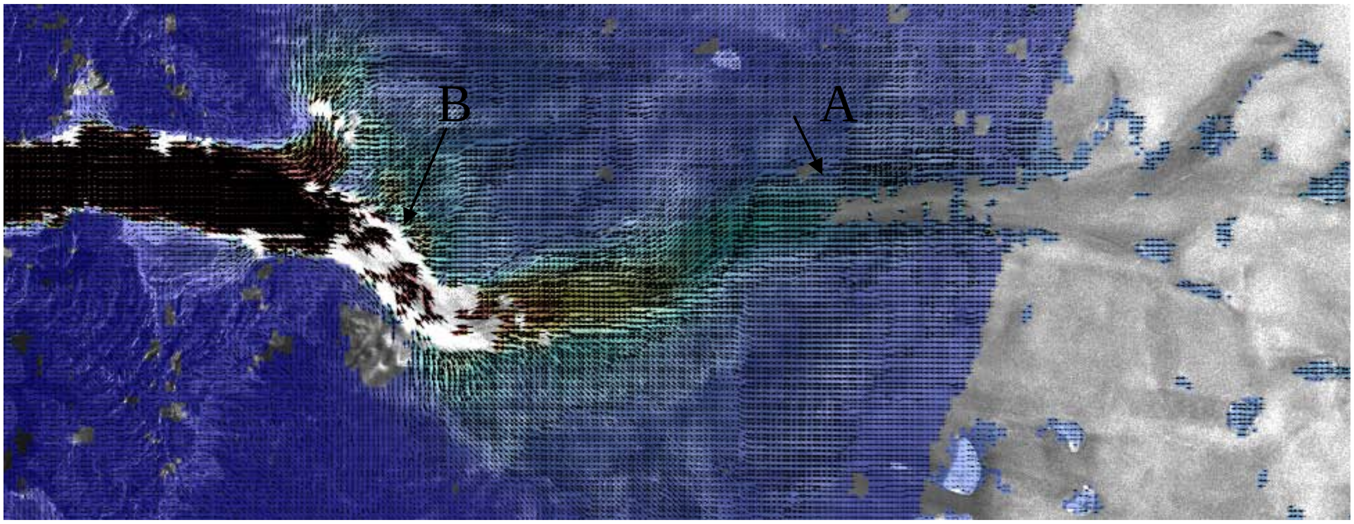


Figure 12. 2006 velocity coverage map depicting areas of sparse velocity coverage within the glacier main drainage. (A) Upper flow regime (B) forward bend.

Finally magnitudes were plotted along the velocity profile line (Figures 13-16). A polynomial line of order 3 was best-fitted to each plot.

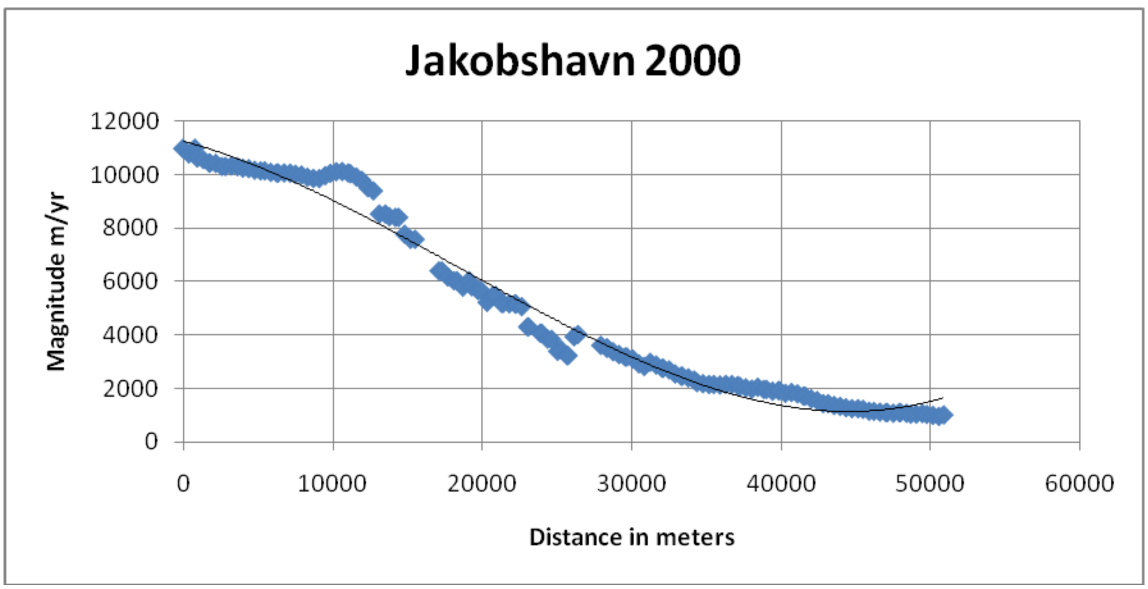


Figure 13. Velocity profiles. Distance along profile line is in meters. Velocities at glacial terminus: 2000 10,996m/yr.

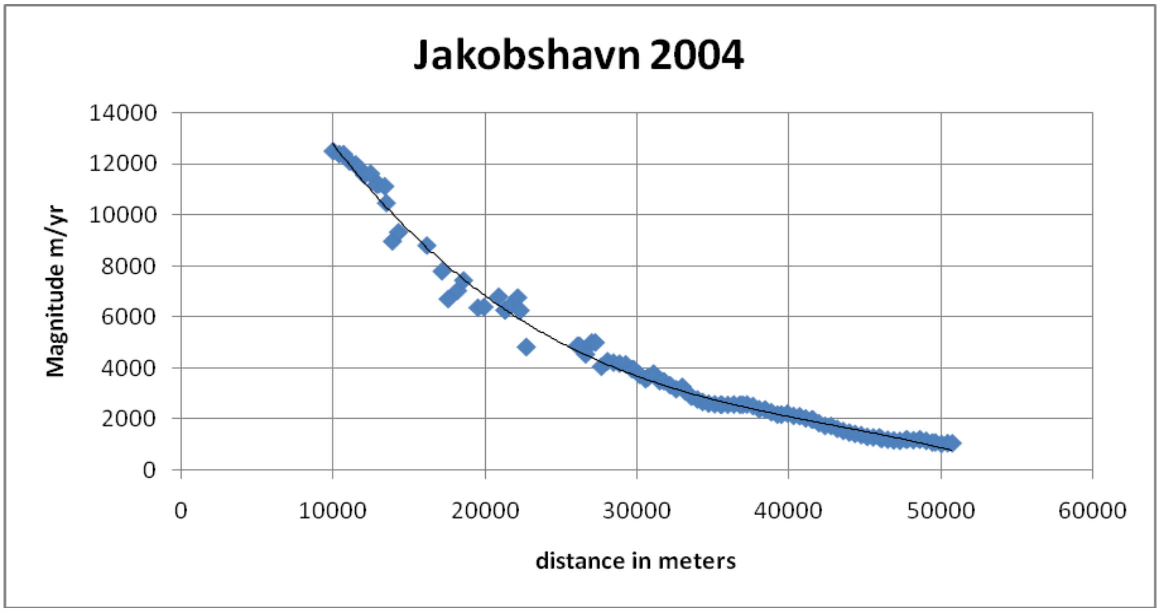


Figure 14. Velocity profiles. Distance along profile line is in meters. Velocities at glacial terminus: 2004 12,505m/yr.

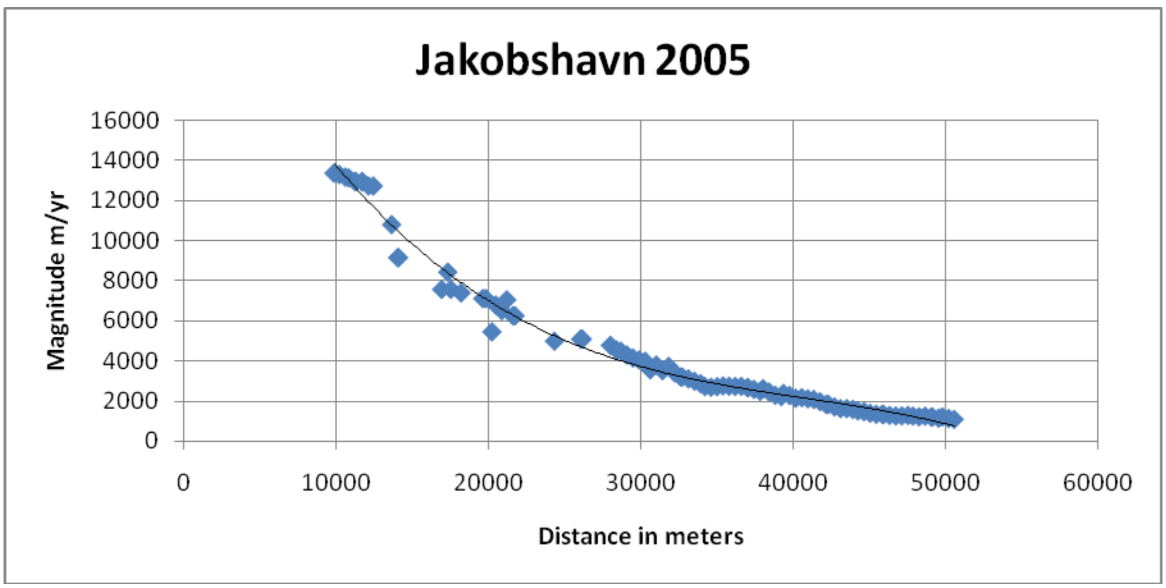


Figure 15. Velocity profiles. Distance along profile line is in meters. Velocities at glacial terminus: 2005 13,341m/yr.

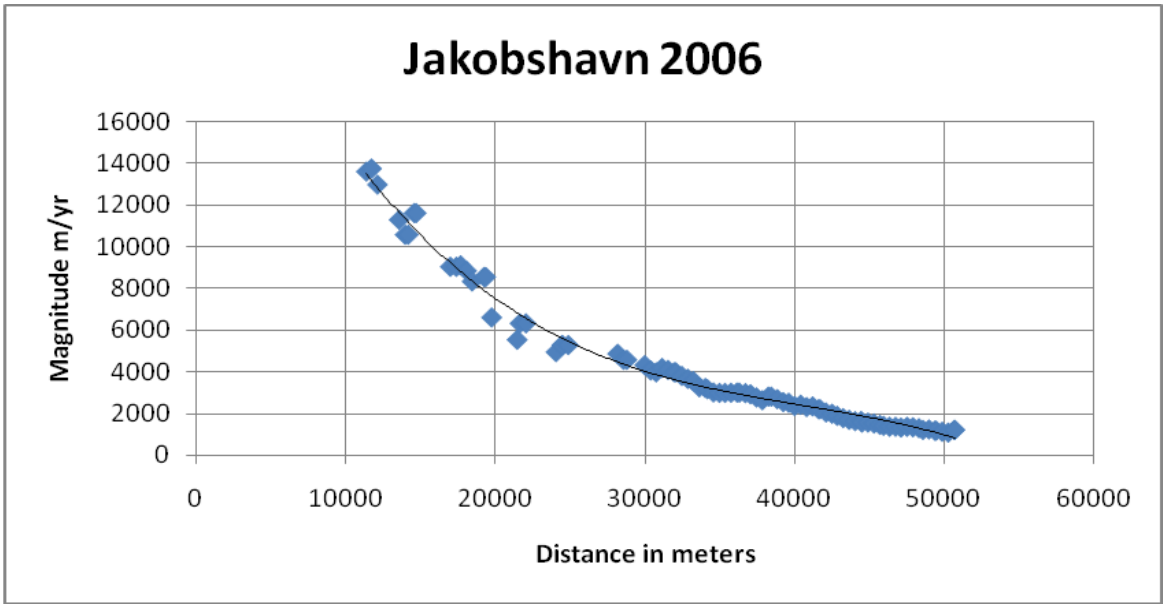


Figure 16. Velocity profiles. Distance along profile line is in meters. Velocities at glacial terminus: 2006 13,771m/yr.

To better compare velocity trends the magnitudes at specific upstream points were plotted (Figure 17) beginning with the glacier terminus in 2006. The velocity field is steadily increasing for each year. There is a divergence in the acceleration between 2000 and 2004-6 at the 42 km mark. Similarly, there is a noticeable change in acceleration between 2004-5 and 2006 at the 26km mark. There does appear to be an acceleration between 2004 and 2005 close to the terminus (~16km mark).

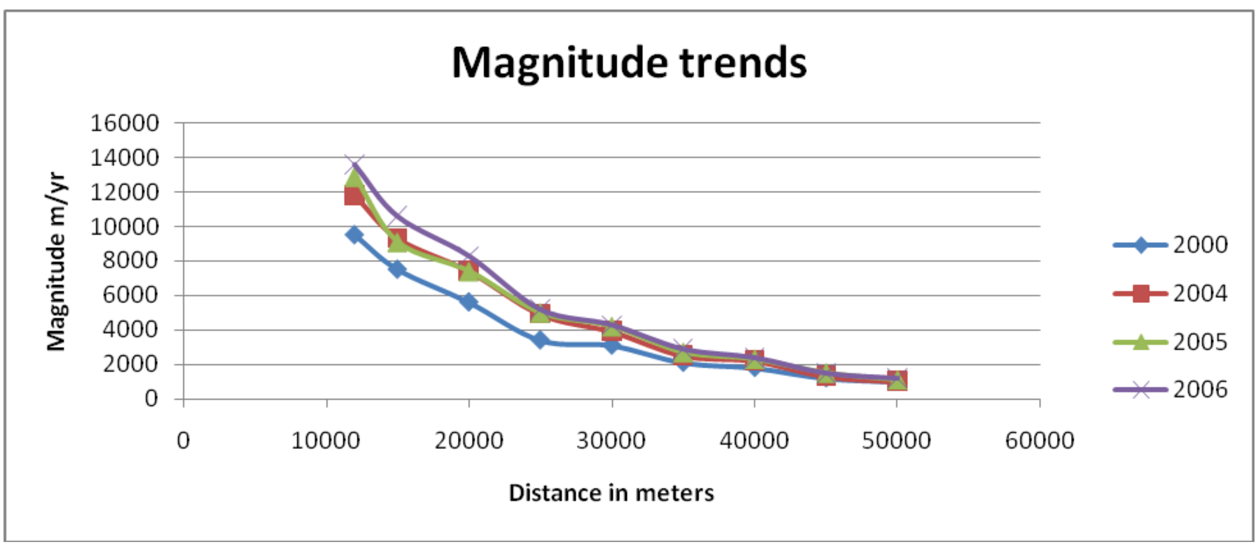


Figure 17. Magnitude trends plotted at specific distance intervals.

Addendum I: Procedure for Creating Final Filtered Velocities from Multiple RAMS Velocity Products

As described in the preceding pages, multiple parameters were used in order to capture the diverse velocity field of the Jakobshavn Glacier Basin. Because of this, multiple versions (or subsets) were created for any single frame pair (repeat pass pair). These instructions outline the steps needed to create, plot, edit and concatenate the magnitude and directional flow arrows for the various RAMS Velocity Products created.

Software used includes: ERDAS Imagine, Matlab, ARCMap, Microsoft Word Pad and Excel. Data are the RAMS Velocity Products. This procedure manual assumes some basic knowledge of each of the software systems listed above. RAMS Velocity Products consist of a README file, a parameter file, five data files, and the source code for two programs to access the data. The parameter file contains the names of the data files, image dimensions, upper left corner map coordinates and pixel spacing. The *.dat file is a binary file containing 8 different elements for each velocity point. Each element is a 32 bit float raster file. The elements consist of magnitude, elevation, east and west component of terrain slope, east, north and vertical component of ice velocity, and speed error. There is also a power image, DEM, and metadata (conflation file with index) available. RAMP Velocity Products are available at the following website for free download: www.bprc.osu.edu/rsl/radarsat/data

*Note: I've embedded a naming convention as an example/guideline. A summary of the directory structure and naming convention is included at the end of this document.

Step 1: Extracting the magnitude and east (Vx) and north (Vy) component of ice velocity from the RAMS Velocity Product data file.

- Compile source code: 'm_data2raster.c'   cc –o m_data2raster m_data2raster.c
- Run code to extract desired data:

  m_data2raster –in <velocity_overview.par> -out <output_filename> -field_name

  field_names: -magnitude, -Vx, -Vy

  naming convention: MAG_MonthYear_FID ie: MAG_NovDec04_Sub1_Zoom1
  Vx_MonthYear_FID
  Vy_MonthYear_FID

  the FID is a unique name given to a particular frame pair and its subsequent subsets.

Step 2: Georeferencing and converting the raster files to an ascii/text file.

- Import Magnitude, Vx, and Vy raster files into ERDAS Imagine.
  - Data Type: IEEE 32 Bit Float
  - Swap Bytes if working from a PC
  - # rows and # cols can be found in the velocity_overview.par file

- Assign projection information
  - Upper Left X, Y can be found in the velocity_overview.par file
    Units are in meters,
    Projection is Polar Stereographic,
    Spheroid is WGS 84,
    Longitude below pole of map is 45W,
    Latitude of true scale is 70N
- Convert pixels to ascii
  - from the main menu bar: UTILITIES-> Convert Pixels to ASCII
    Input filenames (1) MAG_*, (2) Vx_*,(3) Vy_*
    Click on ADD to actually add them one at a time (Output name: pix2ascii.asc)

Step 3: Calculating flow direction (angle of flow).

- Open the pix2ascii file in WordPad or NotePad and delete all the header information, save.
- To delete all the no-data entries, open MATLAB from your current working directory and run the following code. This will create a file with only viable data (pix2ascii_no0.asc)

```
A=load('pix2ascii.asc');
[I]=find(A(:,5)~=0);

fid=fopen('pix2ascii_no0.asc','w');
for i=1:length(I)
    fprintf(fid,'%8.3f\t          %8.3f\t          %f\t          %f\t
%f\n',A(I(i),1),A(I(i),2),A(I(i),3),A(I(i),4),A(I(i),5));
end
fclose(fid);
```

- Open the pix2ascii_no0 file in EXCEL spreadsheet. Insert a row on top and add the following column headings: X Y mag vx vy angles
- To calculate the angles (flow direction) use the following equation:

=degrees(ATAN 2 (vx, vy))

- Copy and special paste resulting column as ‘VALUES’ and delete the formula column
- Save as *.txt file

Step 4: Creating an ARC shapefile.

- In ARCMap click on the + (add data) icon (pix2ascii_no0.txt)
- Right click on the file name to pull up a menu. Select “Display X,Y Data” to create an *.events file (pix2ascii_no0.events)

- Right click on the *.events file to pull up a menu. Select "Data->Export Data (as a shapefile). (Output filename: mag_angles.shp)

Step 5: Filtering out 'bad' values. First filter out any velocity vectors pointing in the opposite direction of glacial flow. For this dataset it's from left to right in the image. For other datasets this may not be applicable.

- Open the mag_angles shapefile and overlay it on the MAG_MonthYear_FID raster file in ARCMap.
- Click on: Selection->Attributes. To get rid of vectors pointing in the wrong direction:
  Vx <0 click on OKAY
- Select "Data-> EXPORT DATA" (shapefile) (Output name: mag_angles_filt.shp)

Next manually remove all offending data points.

- Use the 'identify' tool in ARC and click on the offending data point. Write down the FID number and continue to the next offending data point and so on.
- Export the attributes file. Right click on the mag_angles_filt.shp file and select 'Attributes'. An attributes table will appear. At the bottom click on 'Options'->Export. Open the 'output table' and change the 'Save as Type' to text file. (Output name: pix2ascii_filt.txt)
- Open the ascii2pix_filt.txt file in EXCEL. Add a FID column and **staring with 0** (zero) create a sequential value for each row (0, 1, 2, 3, 4….). Copy and special paste as VALUE in a second column. Delete formula column.
- Search/Find the row with the corresponding FID number of a 'bad' data pixel and delete that row. When finished, delete the FID column and save as a text file. (Output name: pix2ascii_filt_1.txt) Repeat Steps 4-5 until all offending points are removed.

Step 6: Creating arrows from the mag_angles_filt_x.shp file.

- Double left click on the shape file to bring up the 'Layer Properties' dialog.
- Show: Quantities -> Graduated Symbols
- Fields: Value -> mag
- Advanced -> Rotation. Rotate Points by Angle in this field: angles. Click on Arithmetic to select the correct orientation
- Template -> Properties -> Type -> Character Marker Symbol. Font -> ESRI Dimensioning. Click on the upward pointing arrow ⇧
- At this point you can add the magnitude file. To display with a color ramp, double click on the 'mag_monthyear.img' file (created in step 2). Select the red to blue rainbow color ramp and invert so that blue is the lowest value and red is the highest. Repeat steps 1 – 5 for all subsets/datasets.

*Note that the magnitude file does not represent the filtered shapefile. This will be resolved when we concatenate all the different subsets/datasets and create a final velocity.

Step 7: Concatenating multiple datasets to form a final filtered velocity product.

- In ARCMap: Spatial Analyst Tool -> Convert -> Features to Raster
  - Input features: filtered shape file
    Field: Vx (and again for Vy)
    Output Cell Size: 400
    Output Raster: change the 'type' to ERDAS IMAGINE. (Output filename: Vx_Sub#_Zoom# and Vy_Sub#_Zoom#)
    Repeat for all the different datasets
- In ERDAS Imagine: Data Prep -> Mosaic Images -> Mosaic Tool
  Edit -> Add Images. Add all the Vx_*.img files
  Set Overlap Function: Average
  Process -> Run Mosaic (Output name: vx_mosaic and vy_mosaic)
  Repeat for all the Vy_*.img files.
- At this point we need to create a concatenated (and filtered) magnitude file. We've developed a model in ERDAS to help us do this. The input files are the vx_ and vy_mosaic files. The output is a 'mag_mosaic' file. The function is:

```
FUNCTION {
        ID 3;
        TITLE "FLOAT";
        POSITION 2.55556, 3.07778;
        VALUE "FLOAT(SQRT (($n1_vx_mosaic POWER 2+$n2_vy_mosaic POWER 2)))";
        AREA UNION;
        CHILD 4;
```

- Convert pixels to ascii
  - from the main menu bar: UTILITIES-> Convert Pixels to ASCII
    Input filenames (1) mag_mosaic, (2) vx_mosaic ,(3) vy_mosaic
    Click on ADD to actually add them one at a time (Output name: pix2ascii.asc)
- Repeat Steps 3 – 5 but omit the filtering step.

Directory structure where 'subset' is one of x-number of adjacent frame pairs for a particular year and 'zoom' is the sequential number of RAMS iterations using different AOI's and parameters needed to capture the various dynamics of the velocity field.

'Name of glacier'/'monthyear'/final_product
/subset_1

/subset_1_zoom_1
/subset_1_zoom_2
/subset_1_zoom_3
/subset_2
/subset_2_zoom_1
/subset_2_zoom_2
/subset_2_zoom_3

Naming convention:

mag_monthyear	RAMS velocity product and ERDAS Imagine Import files
vx_monthyear	
vy_monthyear	
ascii2pix	ASCII output
ascii2pix_no0	ASCII output removing no-data values
ascii2_pix_filt_x	ASCII output with manually removed ‘bad’ data points
mag_angle	ARC shapefile
mag_angle_filt_x	Filtered ARC shapefile
vx_subset#_zoom#	Filtered IMAGINE files for final concatenation (final_product)
vy_subset#_zoom#	
mag_subset#_zoom#	
vx_mosaic	Final filtered concatenated (mosaicked) files
vy_mosaic	
mag_mosaic	

Below is a list of all the RAMS Velocity Products produced. These products can be found on the RAMS machine (mamm) in the directory /amm/missions/radsat_greenland/ifr/overviews. They can also be found on the Remote Sensing Lab general directory /data/x/Jakobshavn/vel_products/overviews/

1 ov_1 "Ov1_NovDec_06"
8 ov_8 "Ov8_subset_test9"
9 ov_9 "Ov09_Sub1_NovDec05"
10 ov_10 "Ov10_Sub1_NovDec05"
11 ov_11 "Ov11_Sub3_NovDec05"
12 ov_12 "Ov12_Sub4_NovDec05"
13 ov_13 "Ov13_Sub1_NovDec06_B"
14 ov_14 "Ov14_Sub1_NovDec04"
15 ov_15 "Ov15_Sub4a_NovDec05"
16 ov_16 "Ov16_Sub2_NovDec06_B"
17 ov_17 "Ov17_NovDec04_Sub2"

18	ov_18	"Ov18_Sub3_NovDec05"
19	ov_19	"Ov19_NovDec04_Sub3"
20	ov_20	"Ov20_NovDec05_Sub4"
21	ov_21	"Ov21_NovDec06B_Zoom1"
22	ov_22	"Ov22_NovDec06B_Zoom2"
23	ov_23	"Ov23_NovDec06B_Zoom3"
24	ov_24	"Ov24_NovDec06B_Zoom4"
25	ov_25	"Ov25_NovDec06B_Zoom5"
26	ov_26	"Ov26_Sub3_NovDec06B"
27	ov_27	"Ov27_NovDec06B_Zoom6"
28	ov_28	"Ov28_NovDec05_Zoom1"
29	ov_29	"Ov29_NovDec04_Zoom1"
30	ov_30	"Ov30_NovDec04_Sub2_zoom2"
31	ov_31	"Ov31_NovDec_zoom3:
32	ov_32	"Ov32_NovDec06B_Zoom7"
33	ov_33	"Ov33_NovDec05_Zoom2"
34	ov_34	"Ov34_NovDec04_Zoom4"
35	ov_35	"Ov35_NovDec05_Zoom3"
36	ov_36	"Ov36_NovDec06_Zoom8"
37	ov_37	"Ov37_NovDec04_Zoom5"
38	ov_38	"Ov38_NovDec04_Zoom6"
39	ov_39	"Ov39_NovDec05_Zoom4"
40	ov_40	"Ov40_NovDec04_Zoom7"
31	ov_41	"Ov41_NovDec05_Zoom5"
42	ov_42	"Ov42_NovDec05_Zoom6"
43	ov_43	"Ov43_NovDec04_Zoom8"
44	ov_44	"Ov44_NovDec06B_Zoom8"
45	ov_45	"Ov45_NovDec05_Zoom7"
46	ov_46	"Ov46_NovDec00_Sub1"
47	ov_47	"Ov47_NovDec00_Sub2"
48	ov_48	"Ov48_NovDec06B_Sub1a"
49	ov_49	"Ov49_NovDec06B_Sub2a"
50	ov_50	"Ov50_NovDec05_Sub2a"
51	ov_51	"Ov51_NovDec05_Sub1A"
52	ov_52	"Ov52_NovDec04_Sub1A"
53	ov_53	"Ov53_NovDec04_Sub2A"
54	ov_54	"Ov54_NovDec06B_Sub2_a"
55	ov_55	"Ov55_Subset_0"
56	ov_56	"Ov56_Subset0_Zoom1"
57	ov_57	"Ov57_Subset0_Zoom2"

58 ov_58 "Ov58_NovDec00_sub3"

A RAMS Velocity Product comprises the following.

-Binary Data File containing:

- magnitude
- elevation
- east component of ice velocity
- north component of ice velocity
- vertical component of ice velocity
- east component of terrain slop
- west component of terrain slope
- speed error
- directional error

-DEM (400m 16-bit)

-Image overview (400m 16-bit)

-Conflation and Index files containing metadata on all source data contributing to a point

- Map coordinates
- Elevation
- East & West component of slope
- East, North, Vertical component of velocity
- Speed and direction error
- Beam mode
- Reference Orbit number, acquisition date
- Secondary Orbit number, acquisition date
- Baseline
- Doppler overlap
- Number of VGCPs
- Followed by a list of the VCPs

-Utility Programs

- To extract binary data files
- To extract metadata

Addendum II: Procedure for Creating VCP (Velocity Control Points) to Propagate From One Overlapping Frame Pair to the Next

Since we have very little velocity control for Greenland other than static features (rock outcroppings), the need to propagate velocities calculated from one frame pair to the adjacent overlapping frame pair is necessary. The input file needed is the mag_angles_filt.shp file created Steps 1 through 4 in addendum 1.

Step 1: In ARCMap add the mag_angles_filt.shp file

- Define projection information: ArcToolbox -> Data Management Tools -> Projections and Transformation -> Define Projection
  - Input Dataset or Feature Class: mag_angles_filt.shp
  - Coordinate System -> import -> mag_month-year.img (use the projection information from and existing ERADAS Imagine file you created in Step 2 of Addendum 1
- Right click on the image in the View (right side) -> Data Frame Properties -> Predefined -> Geographic Coordinate Systems -> World -> WGS84 This will reproject the Polar Stereographic map projection into Latidute/ Longitude.
- Right click on the mag_angles_filt layer (left side) -> Export -> **The Data Frame** -> Output Filename (ie: vcp_monthyear) This will give you a shapefile in lat/long. Make sure you choose the 'data frame' option!
- Go to Attributes table and add a longitude and latitude field (options->add field)

  Name: long Name: lat
  Type: float Type: float
  Precision: 9 Precision: 9
  Scale: 5 Scale: 5

  Right click on longitude column. Go to Field Calculator and use this function:

```
Dim Output As Double
Dim pPoint As IPoint
Set pPoint = [Shape]
Output = pPoint.X
__esri_field_calculator_splitter__
Output
```

  Use pPoint.X for longitude and pPoint.Y for latitude
- Export as ASCII file. In Attributes table->option->EXPORT -> vcp_monthyear.csv
  Save Type: **Text File**

Step 2: Next we need to get the exported ascii file into a format that is compatible with the RAMS VphaseTool VCP ingest function.

- Open file in EXCEL and add a new column heading: new_angle. We need to calculate the flow direction (new_angle) with respect to the new projection grid (lat/long). To do this we need to:

  1. Calculate the true north direction:
     (longitude below pole of map: -45 degree) – (longitude)

  2. Calculate the north angle with respect to the x-axis
     (90 degrees) – (true north direction)

  3. Calculate the vector angle with respect to the x-axis:
     (north angle) – heading *heading being the ‘angles’ column

     new_angle = +90 - (-45-longitude) – angle (Greenland)
     new_angle = +90 – longitude – angle (Antarctica)

- Arrange data in the spreadsheet in the following order:
  lat long mag new_angle
  delete all other columns and the header row. Save as *.csv file.

Step 3: Next we need to convert our new *.csv file into a RAMS friendly format. To do this we’ve written the following UNIX awk script:

```
#!/bin/sh
Awk ‘FS=”,” { print NR-1 ”:”, $1, $2, “n/a”, $3, $4, “0  na   na   na   na   na
MonthYear  Year  Year  VGCPs”}’ $1
```

Where the Month and Year correspond to the month and year of the related dataset. When running this script, pipe the output to a file with a *.txt extension ie: vcp_monthyear.txt

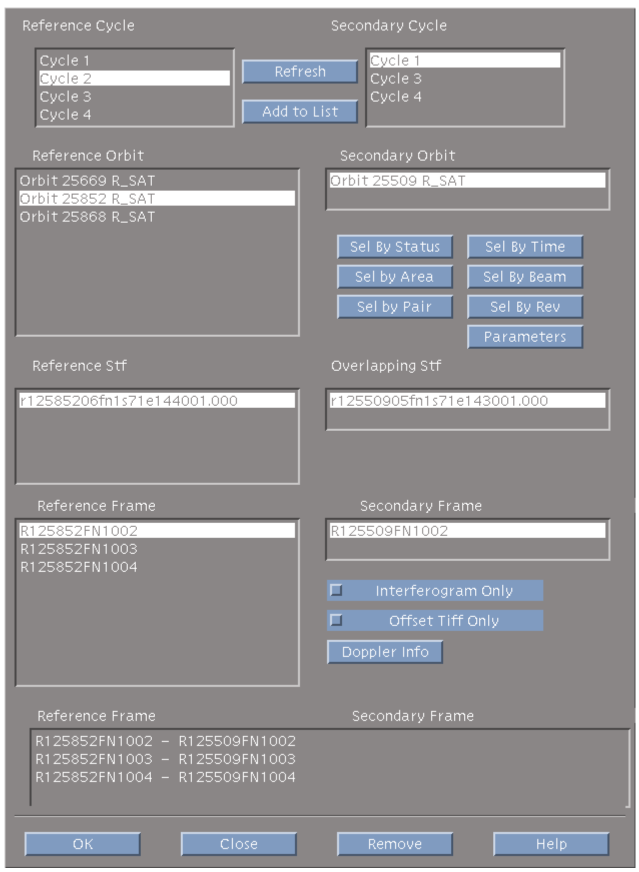
- Lastly, open the text file in a text editor and add the following first line:
  number_of_vcps: #####

Addendum III: RAMS PHASE I PROCESSING

Phase I processing consists of two operations, registering the two SLC frames and creating the interferogram. Typically both operations are done in one step.

Pair Selection Dialog

The figure below shows the dialog used to select SLC frame pairs to be processed. Frame pairs are selected by cycle, orbit and STF. The “Add to List” button adds the currently selected frame pair to the pairs list. The “Remove” button removes a frame pair. Upon pushing OK, all frame pairs in the list will have interferograms created. If only interferograms are needed to be generated, the “Interferogram only” toggle is enabled. The “Offset Tiff Only” button allows the generation of a set of tiffs containing the range and azimuth components of the offset.



Frame Pair Selection Dialog.

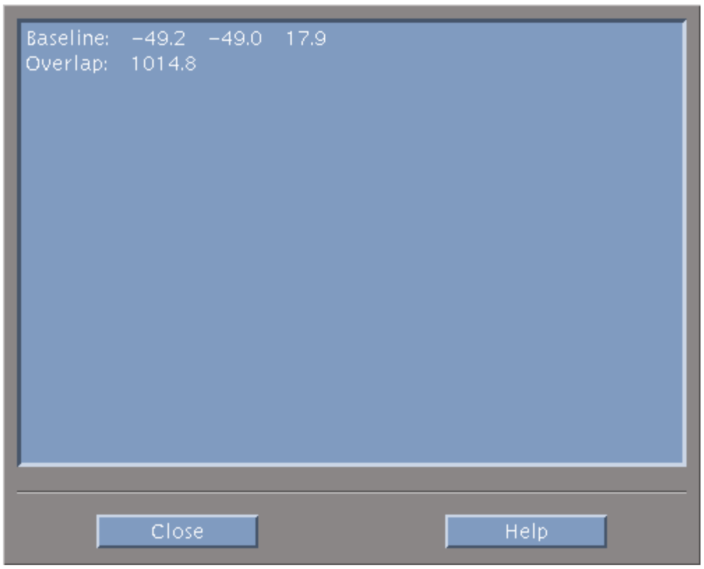
It is convenient to restrict the list of frames in order to speed up drawing of the frames in the main window. The following options are available to extract a subset of the IFR frame lists:

“Sel By Status” allows selection of frames based on the database status.

“Sel By Time” allows selection of frames in a given time interval.
“Sel By Area” allows the user to restrict pairs to a selected area on the main drawing window .
“Sel By Beam” allows the selection of pairs of a single beam mode.
“Sel By Pair” Allows restricting the list to a specified cycle 0combination.
“Sel By Rev” allows the user to select from a specific range of revolution numbers.

Most of the selection dialogs have a similar interface.

Doppler info button allows the user to look at the Doppler and baseline parameters for a given frame before spawning the job.



Doppler Information Display

Optional parameters are available through the Parameters button. All option defaults are stored in the $RAM_MISSION/VphaseDefaults file. The values in the GUI are initialized from this file and the options are passed to the individual programs through Parameter files.

The following is the current Defaults File:

```
processors        : 1

regist {
  over_sampling_factor    :   1
  max_line_offset         : 2000
  use_tie_points          :  off
  corr_mode               :   3

  offset {
   line_offset            :   0
   pixel_offset           :   0
```

```
  }

 xform {
  xline_search_radius    :   35
  xpixel_search_radius   :   27
  xline_chip_size        :   64
  xpixel_chip_size       :   64
  xline_spacing          : 1021
  xpixel_spacing         :  251
  xbad_coherence         : 0.12
  xlower_good_coherence  : 0.15
  xline_chip_size2       :  128
  xpixel_chip_size2      :   64
  xbad_coherence2        : 0.12     }
 offsetmap {
  line_search_radius    :    8
  pixel_search_radius   :    4
  line_spacing          :  160
  pixel_spacing         :  80
  line_chip_size        :   64
  pixel_chip_size       :   64
  bad_coherence         : 0.12
  good_coherence        : 0.50
  lower_good_coherence  : 0.15
  line_chip_size2       :  128
  pixel_chip_size2      :   64
  bad_coherence2        : 0.12
  }
}

create_interferogram_coherence {
 use_affinexform          :  off
 local_slope_mode         :  on
 bad_coherence_threshold  :  0.12
 range_looks              :  2
 along_track_looks        :  8
 range_window             :  3
 along_track_window       :  3
 height                   :  0.0
 make_mode                :  resample_make
 bad_coherence_threshold2 :  0.12
 used_item_percent        : 10.0
 no_filter                :  0
 mwl                      :  5
 }
```

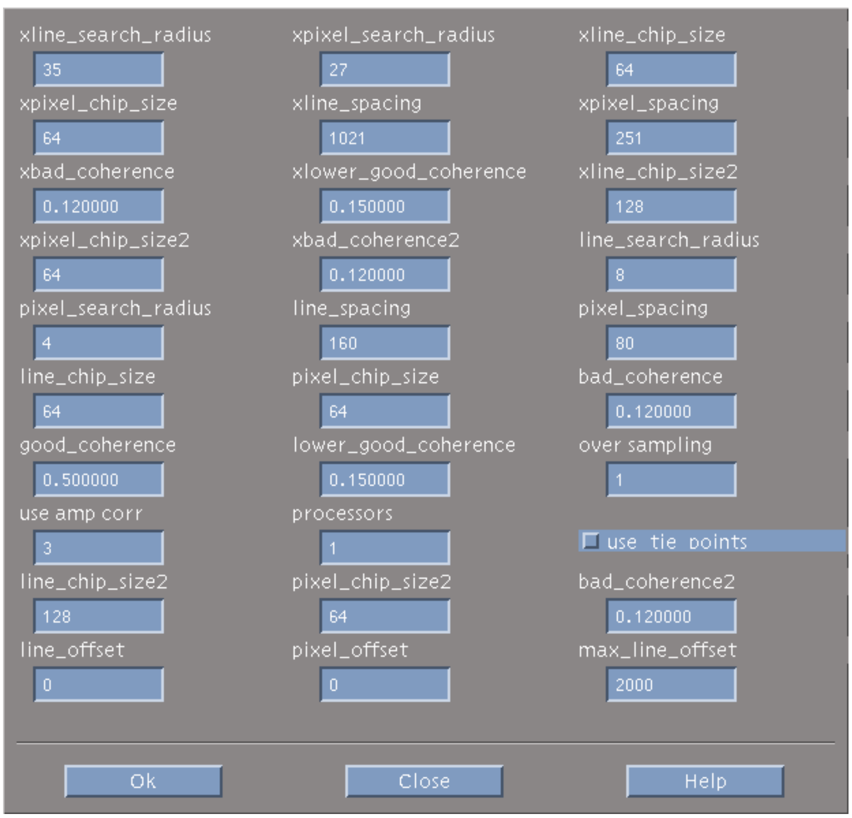
```
create_offsetmap_tifs {
   -mode : 0
   -coherence_th: 0.12
   -remove_linear_term: 1
   }
```

dem_data   : /$RAMS_DEM/vexcel200m.dem
dem_data_par   : /$RAMS_DEM/vexcel200m.dem.hdr

Parameters are grouped according to the applicable program. For each parameter, the location of the default value is stored in a defaults file located in $RAMS_MISSION. The variables are grouped according to the executables using them. The values are specified by which Defaults file it is located in and which group and tag name under which the variable is found. Thus the line;

default location: VphaseDefaults - regist / xform / xline_search_radius
indicates the VphaseDefaults file, regist group, xform subgroup and the tag in that group is xline_search_radius

If the parameter is optional there is a '-' in front of the tag. The Default Value is the default at the time of writing and can be changed at anytime by modifying the Default File.



Options for regist

Variable: xline_search_radius - integer
default location: VphaseDefaults - regist / xform / xline_search_radius
Current Default: 35
Admissible Values: > 0
Explanation: for setting search window size (2*xline_search_radius + 1) in azimuth direction (line) for tie points for affine transform. Increasing it will increase execution time and also the possibility to find a good match.

Variable: xpixel_search_radius - integer
default location: VphaseDefaults - regist / xform / xpixel_search_radius
Current Default: 27
Admissible Values: > 0
Explanation: for setting search window size (2*xpixel_search_radius + 1) in range direction (pixel) for tie points for affine transform. Increasing it will increase execution time and also the possibility to find a good match.

Variable: xline_chip_size - integer
default location: VphaseDefaults - regist / xform / xline_chip_size
Current Default: 64
Admissible Values: power of 2
Explanation: line chip size for cross correlation to find tie points for affine transform. With smaller chip size you will get more local but noisier match. With bigger chip size you will get more reliable but more global measurements.

Variable: xpixel_chip_size - integer
default location: VphaseDefaults - regist / xform / xpixel_chip_size
Current Default: 64
Admissible Values: power of 2
Explanation: pixel chip size for cross correlation to find tie points for affine transform. With smaller chip size you will get more local but noisier match. With bigger chip size you will get more reliable but more global measurements.

Variable: xline_spacing - integer
default location: VphaseDefaults - regist / xform / xline_spacing
Current Default: 1021
Admissible Values: > 0
Explanation: line spacing of the tie point grid for affine transform.

Variable: xpixel_spacing - integer
default location: VphaseDefaults - regist / xform / xpixel_spacing
Current Default: 251
Admissible Values: > 0
Explanation: pixel spacing of the tie point grids for the affine transform.

Variable: xbad_coherence - float
default location: VphaseDefaults - regist / xform / xbad_coherence
Current Default: 0.12
Admissible Values: > 0.0 but < 1.0
Explanation: coherence threshold for tie points of the affine transform.

Variable: xlower_good_coherence - float
default location: VphaseDefaults - regist / xform / xlower_good_coherence
Current Default: 0.15
Admissible Values: > xbad_coherenence
Explanation: if coherence is bigger than this lower_good_coherence threshold, the tie point will be accepted. Otherwise the SNR will be calculated and compared the preset SRN threshold to determine whether the tie point is accepted or not.

Variable: xline_chip_size2 - integer
default location: VphaseDefaults - regist / xform / xline_chip_size2
Current Default: 128
Admissible Values: power of 2
Explanation: line chip size for second registration method for affine transform tie points.
Variable: xpixel_chip_size2 - integer
default location: VphaseDefaults - regist / xform / xpixel_chip_size2
Current Default: 64
Admissible Values: power of 2.
Explanation: pixel chip size for second registration method for affine transform tie points.

Variable: xbad_coherence2 - float
default location: VphaseDefaults - regist / xform / xbad_coherence2
Current Default: 0.12
Admissible Values: > 0.0 but < 1.0
Explanation: coherence threshold for the second registration method for tie points of the affine transform.

Variable: line_search_radius - integer
default location: VphaseDefaults - regist / offsetmap / line_search_radius
Current Default: 8
Admissible Values: > 0
Explanation: for setting search window size (2*line_search_radius + 1) in azimuth direction (line) for tie points of OffsetMap. Increasing it will increase execution time and also the possibility to find a good match.

Variable: pixel_search_radius - integer
default location: VphaseDefaults - regist / offsetmap / pixel_search_radius
Current Default: 4
Admissible Values: > 0

Explanation: for setting search window size (2*pixel_search_radius + 1) in range direction (pixel) for tie points of OffsetMap. Increasing it will increase execution time and also the possibility to find a good match.

Variable: line_spacing - integer
default location: VphaseDefaults - regist / offsetmap / line_spacing
Current Default: 160
Admissible Values: > 0
Explanation: line spacing of the tie point grids for OffsetMap.

Variable: pixel_spacing - integer
default location: VphaseDefaults - regist / offsetmap / pixel_spacing
Current Default: 80
Admissible Values: > 0
Explanation: pixel spacing of the tie point grids for OffsetMap.

Variable: line_chip_size - integer
default location: VphaseDefaults - regist / offsetmap / line_chip_size
Current Default: 64
Admissible Values: power of 2
Explanation: line chip size for registration to create OffsetMap tie points.

Variable: pixel_chip_size - integer
default location: VphaseDefaults - regist / offsetmap / pixel_chip_size
Current Default: 64
Admissible Values: power of 2.
Explanation: pixel chip size for registration to create OffsetMap tie points.

Variable: bad_coherence - float
default location: VphaseDefaults - regist / offsetmap / bad_coherence
Current Default: 0.12
Admissible Values: > 0
Explanation: coherence threshold for OffsetMap tie points.

Variable: good_coherence - float
default location: VphaseDefaults - regist / offsetmap / good_coherence
Current Default: 0.50
Admissible Values: > lower_good_coherence
Explanation:

Variable: lower_good_coherence - float
default location: VphaseDefaults - regist / offsetmap / lower_good_coherence
Current Default: 0.15
Admissible Values: > bad_coherence

Explanation: if coherence is bigger than this lower_good_coherence threshold, the tie point will be accepted. Otherwise the SNR will be calculated and compared the preset SRN threshold to determine whether the tie point is accepted or not.

Variable: over_sampling_factor - integer
default location: VphaseDefaults - regist / over_sampling_factor
Current Default: 4
Admissible Values: > power of 2
Explanation: over sampling factor to over sample the data of a chip.

Variable: corr_mode (use_amp_corr) - boolean
default location: VphaseDefaults - regist / corr_mode
Current Default: 3
Admissible Values: 0,1, 3.
0: complex registration mode;
1: amplitude registration mode;
3: combination mode with complex and amplitude registration.
Explanation: registration mode option for OffsetMap tie point generation.

Variable: processors - integer
default location: VphaseDefaults - processors
Current Default: 1
Admissible Values: 1 – the number of processors available.
Explanation: option for multiple processors

Variable: use_tie_points - boolean
default location: VphaseDefaults - regist / use_tie_points
Current Default: off
Admissible Values: off / on
Explanation: If it is on, the tie points included in the .tie file will be used to create the affine transform. If it is off, the default procedure will apply, i.e., register for tie points for affine transform and then create the affine transform and finally search for tie points of OffsetMap based on the affine transform.

Variable: line_chip_size2 - integer
default location: VphaseDefaults - regist / offsetmap / line_chip_size2
Current Default: 128
Admissible Values: >= line_chip_size
Explanation: line chip size for the second registration method (amplitude method) for OffsetMap generation.

Variable: pixel_chip_size2 - integer
default location: VphaseDefaults - regist / offsetmap / pixel_chip_size2
Current Default: 64
Admissible Values: >= pixel_chip_size

Explanation: pixel chip size for the second registration method (amplitude method) for OffsetMap generation.

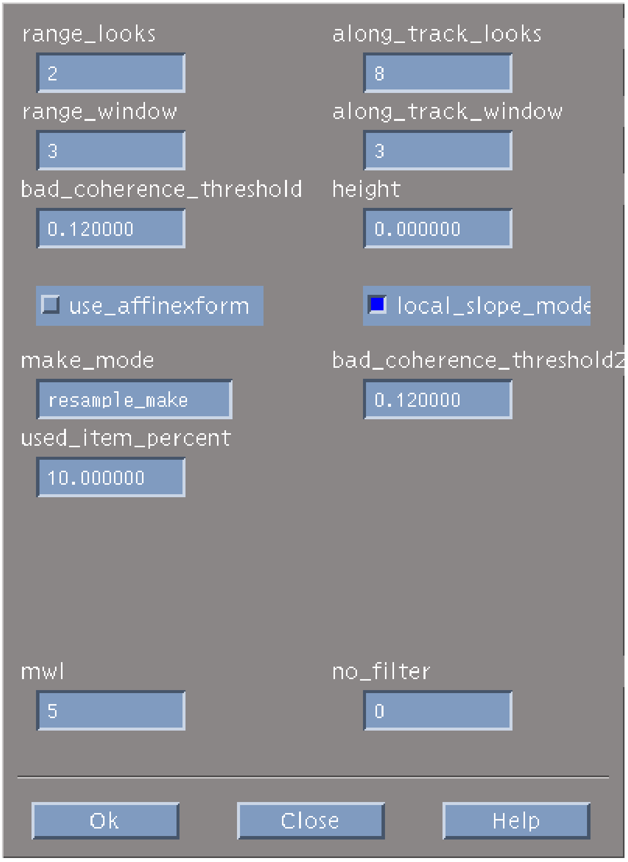
Variable: bad_coherence2 - float
default location: VphaseDefaults - regist / offsetmap / bad_coherence2
Current Default: 0.12
Admissible Values: > 0
Explanation: coherence threshold for the second registration method for OffsetMap generation.

Variable: line_offset - integer
default location: VphaseDefaults - regist / offset / line_offset
Current Default: 0 (not active)
Admissible Values: no limit
Explanation: This is used if the registration needs to be seeded using an initial line offset. This would be used if the ephemeris is far enough off to cause registration to fail.

Variable: pixel_offset - integer
default location: VphaseDefaults - regist / offset / pixel_offset
Current Default: 0
Admissible Values: no limit
Explanation: This is used if the registration needs to be seeded using an initial pixel offset. This would be used if the ephemeris is far enough off to cause registration to fail.

Variable: max_line_offset - integer
default location: VphaseDefaults - regist / max_line_offset
Current Default: 2000
Admissible Values: no limit
Explanation: If the line offset determined by ephemeris is bigger than this, it will throw an exception because of the poor overlap. Reframing may be necessary.

Options for create_interferogram_coherence



Variable: range_looks - - integer
default location: VphaseDefaults - create_interferogram_coherence / range_looks
Current Default: 2 for standard beams; 3 for fine beams.
Admissible Values: > 0
Explanation: number of looks in range direction for decimation and averaging of the interferogram.

Variable: along_track_looks - integer
default location: VphaseDefaults - create_interferogram_coherence / along_track_looks
Current Default: 8 for standard beams; 3 for fine beams.
Admissible Values: > 0
Explanation: number of looks in azimuth direction for decimation and averaging of the interferogram.

Variable: range_window - integer
default location: VphaseDefaults - create_interferogram_coherence / range_window
Current Default: 3
Admissible Values: > 0

Explanation: extra averaging window in range for creating coherence image out of the interferogram.

Variable: along_track_window - integer
default location: VphaseDefaults - create_interferogram_coherence / along_track_window
Current Default: 3
Admissible Values: > 0
Explanation: extra averaging window in azimuth for creating coherence image out of the interferogram.

Variable: bad_coherence_threshold - float
default location: VphaseDefaults - create_interferogram_coherence / bad_coherence_threshold
Current Default: 0.12
Admissible Values: > 0
Explanation: coherence threshold to cut off the tie points.

Variable: height - float
default location: VphaseDefaults - create_interferogram_coherence / height
Current Default: 0.0
Admissible Values: reasonable height
Explanation: nominal elevation for the whole frame of data.

Variable: use_affinexform - boolean
default location: VphaseDefaults - create_interferogram_coherence / use_affinexform
Current Default: off
Admissible Values: on / off
Explanation: If it is on, the affine transform instead of the OffsetMap will be used for resampling secondary SLC. This mode is designed for the case that there are very few or no good tie points found but we still want to create the interferogram even without any fringes.

Variable: local_slope_mode - boolean
default location: VphaseDefaults - create_interferogram_coherence / local_slope_mode
Current Default: on
Admissible Values: on / off
Explanation: on: use fringe rate locally estimated to flatten the interferogram before doing multi-looking; off: use ephemeris to flatten the interferogram.
Variable: make_mode - string
default location: VphaseDefaults - create_interferogram_coherence / make_mode
Current Default: resample_make
Admissible Values: resample_only, make_only, resample_make
Explanation: resample_only: resample secondary SLC only without creating interferogram.
make_only: create interferogram using existing resampled SLC.
resample_make: both of resample_only and make_only.

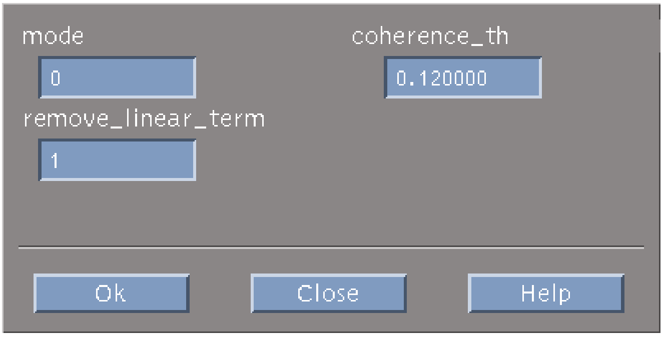
Variable: bad_coherence_threshold2 - float
default location: VphaseDefaults - create_interferogram_coherence / bad_coherence_threshold2
Current Default: 0.12
Admissible Values: > 0
Explanation: coherence threshold for cutting off the amplitude tie points from OffsetMap.

Variable: used_item_percent - float
default location: VphaseDefaults - create_interferogram_coherence / used_item_percent
Current Default: 10.0 %
Admissible Values: 0 ~ 100%
Explanation: For OffsetMap refinement. Only when there is more percentage of similar tie points than this threshold out of the good tie points within a fixed 21 x 21 window, this current tie point will be accepted. Otherwise this tie point will be deleted even if it has higher coherence than bad_coherence_threshold.

Variable: no_filter - boolean
default location: VphaseDefaults - create_interferogram_coherence / no_filter
Current Default: 0
Admissible Values: 0 / 1
Explanation: 0: filter is applied for OffsetMap filtering as usual.
1: no filter will be applied, the .off.use will be directly used for resampling (together with matlab code)

Variable: mwl - integer
default location: VphaseDefaults - create_interferogram_coherence / mwl
Current Default: 5
Admissible Values: > 0
Explanation: window size for moving window averaging for OffsetMap after refinement and triangulation interpolation.

Option for Offset Tiff



Variable: mode - integer
default location: VphaseDefaults - create_offsetmap_tifs / -mode
Current Default: 0
Admissible Values: 0 / 1
Explanation: do (0) or don't (1) create tif images.
Variable: coherence_th - float
default location: VphaseDefaults - create_offsetmap_tifs / -coherence_th
Current Default: 0.12
Admissible Values: > 0
Explanation: coherence threshold for cutting off the tie points for tif images of offsetmap.

Variable: remove_linear_term - boolean
default location: VphaseDefaults - create_offsetmap_tifs / -remove_linear_term
Current Default: 1
Admissible Values: 0 / 1
Explanation: to remove (1) or not (0) the linear term of line offsets before creating tif images of offsetmap.

Addendum IV: RAMS Phase II Processing

Once the interferograms are created, velocities are generated through the following procedure:

1) Mosaic datatake pairs
2) Create Segments
3) Create Cut Files
4) Unwrap Phase
5) Edit Seeds if necessary and repeat 5
6) Add VCP's
7) Create Tidal Model if required
8) Estimate Baseline, repeat 7 & 8 if necessary
9) Remove DEM
10) Estimate Velocities
11) Merge Velocities from crossing orbits.
12) Mosaic Velocities

Phase II processing steps are handled by one of 10 executables. Each executable that is used for phase II processing has a group in that file that contains all values applicable to that executable. Those values are used to initialize the individual Options dialogs where applicable. The following is the current Velocity Defaults file:

```
prl        : 1
log        :

no_data {
   int                        : 32768
   float                      : -1000000.0
   string                     : ""
   }

mosaic_offsetmap {
    -remove_line_offset_streaks   : 0
    -coherence_threshold          : 0.12
    -averaging_length             : 41
   }

filter_Interferogram {
      -looks                  : 1.0
      -fft_length             : 32
      -filter                 : GOLD
   }

segment {
   -segment_length         :  5000
   }

doUnwrapping {
   -unwrapping_mode        :  6
```

```
   -qmask                    :  0.25
   -st                       :  0.5
   -ct                       :  0.1
   -gt                       :  0.5
   -method                   :  WU
  }

removeDEM {
   -tide_mask:     filename_for_tidal_mask_data
   -tide_par:      filename_for_tidal_tiepoints
   -num_coeff                : 3
   -mse                      : 5.0
   -uo                       : 0
   -avr                      : 1
   }

create_velocity_map {
   -type                     : 1
   -cvm_mode                 : 3
   -dem_error                : 10.0
   -tide_mask:  filename_for_tidal_mask_data
   -tide_par:   filename_for_tidal_tiepoints
   }

generate_vgcp {
     -uo                     : 0
   }

generate_cvm  {
    -ex_scale                : 0.35
    -vector_scale            : 0.1
    -arrow_value             : 0
    -dec_x                   : 1
    -dec_y                   : 1
    -no_data_replacement     : 0.0
    -indicator               : 0
}

create_offsetmap_tifs {
    -mode : 0
    -coherence_th: 0.12
    -remove_linear_term: 1
    }

create_velocity_metadata {
    -use_azimuth_offset:   0
    }

published_vcps    :  /$RAMS_MISSION/ifr/vcps/PublishedVcps.txt
balance_vel_dir   :  /$RAMS_MISSION/ifr/vcps/balance_velocity_map.dir
balance_vel_mag   :  /$RAMS_MISSION/ifr/vcps/balance_velocity_map.mag
balance_vel_par   :  /$RAMS_MISSION/ifr/vcps/balance_velocity_map.par
dem_data          :  /$RAMS_DEM/vexcel200m.dem
dem_data_par      :  /$RAMS_DEM/vexcel200m.dem.hdr
grounding_line_data :  /$RAMS_COAST/coast03_new2
```

For all steps, VphaseTool writes out a parameter file to the $RAMS_HOME/tmp directory with the file name velocity* where each parameter file gets a sequential number. The default parameters are written to that file unless the user changes them. A list of datatake pairs is also included in the list. The process is then managed by a call to
rd_velocity –parm $RAMS_HOME/tmp/velocity0* -step_name

rd_velocity writes out another parameter file to each separate directory for each datatake pair and then spawns one or more executables to do the step and updates the database status.

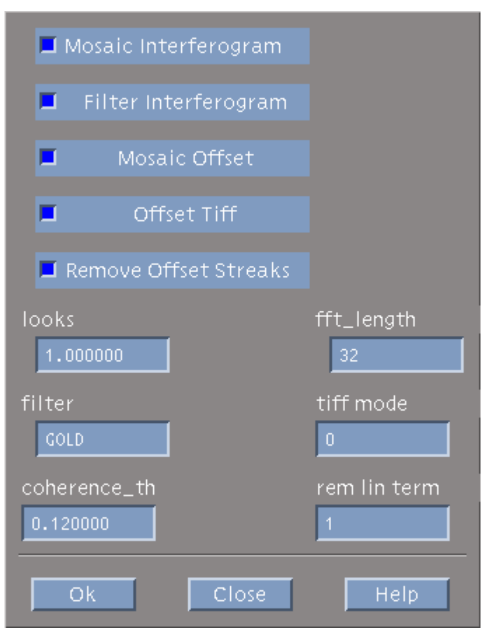
For the editing steps, VphaseTool spawns a call to vView –parm par_file, using the parm file for the appropriate datatake pair.

Step 1: Mosaic STFs

All frame pairs contained in a pair of datatakes are mosaiced together into one long interferogram. The power and offsets are also combined.

Mosaic Interferogram- ON mosaics the individual IFR frames for a cycle pair. This is necessary when mosaicking for the first time or when the SLCs/IFRs had to be reprocessed due to missing lines. Otherwise the option can be turned off.

Filter Interferogram- ON creates a filtered IFR from the moaicked cycle pair created above. This is necessary when mosaicking for the first time or when the SLCs/IFRs had to be reprocessed due to missing lines. Otherwise the option can be turned off.



Mosaic Offset- ON mosaics the individual OFFSETS calculated for each frame of a cycle pair. This is necessary when mosaicking for the first time or when the SLCs/IFRs had to be reprocessed due to missing lines. Leaving this option ON will also over-write any modifications made to the offsets during the ionispheric streak filtering.

Offset Tiff- ON creates a series of TIFF files for viewing across track streaking caused by ionispheric interference.

Remove Offset Streaks- ON This will calculated and apply a global slope interval and filtering. It may be advisable to have this option turned off and the user manually define areas to be filtered. This would be to avoid introducing artifacts in more complicated areas.

Variable: looks - float
default location: VelocityDefaults - filter_Interferogram / -looks
Current Default: 1.0
Admissible Values: > 0
Explanation: Goldstein filter parameter alfa = looks/2. The resulting spectrum from Goldstein filter = (original spectrum)$^{1 + alfa}$. By increasing the value of looks (or the alpha value $1<\alpha<2$) the IFR filtering will be increased. The down side is that coupled with lowering the q-value, more phase unwrapping error may be introduced.

Variable: fft_length - integer
default location: VelocityDefaults - filter_Interferogram / -fft_length
Current Default: 32
Admissible Values: power of 2
Explanation: Higher results in better smoothing but maybe some artifacts.

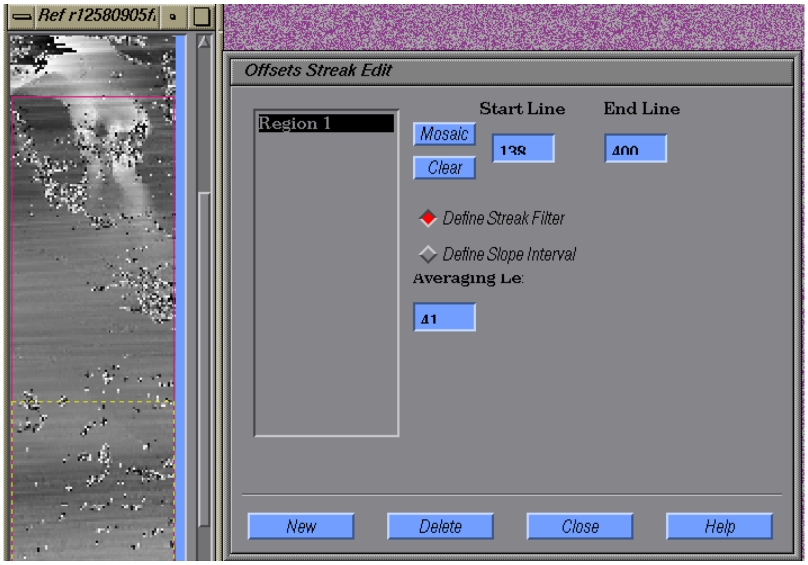
Variable: filter - string
default location: VelocityDefaults - filter_Interferogram / -filter
Current Default: GOLD
Admissible Values: GOLD / DIFF
Explanation: Goldstein (GOLD) or diffusion (DIFF) filter for interferogram filtering

Variable: tiff_mode - integer
default location: VelocityDefaults - create_offsetmap_tifs / -mode
Current Default: 0
Admissible Values: 0 / 1
Explanation: To create (1) or not (0) the tif images of the offsetmap.

Variable: coherence_th - float
default location: VelocityDefaults - create_offsetmap_tifs / -coherence_th
Current Default: 0.12
Admissible Values: > 0
Explanation: To cut off the tie points below the threshold for displaying.

Variable: rem_lin_term - boolean
default location: VelocityDefaults - create_offsetmap_tifs / -remove_linear_term
Current Default: 1
Admissible Values: 0 / 1
Explanation: remove (1) or not (0) linear term of line offset before creating the tif images.

Step 2: Edit Streaks



Define Slope Interval - Draws a box around an area with visible across track streaks to determine an average slope. This can be modified to increase or decrease the area containing streaks.

Define Streak Filter - Draws box(es) around area(s) that show visible across track streaks.

Averaging Length. Increasing this value (default is 41) will apply a more global averaging verses a more local averaging.

The process is basically trial and error until the streaks are smoothed out sufficiently without introducing artifacts elsewhere, however, initial experiments seem to show that a value of about 200 was preferable.

Step 3: Edit Segments

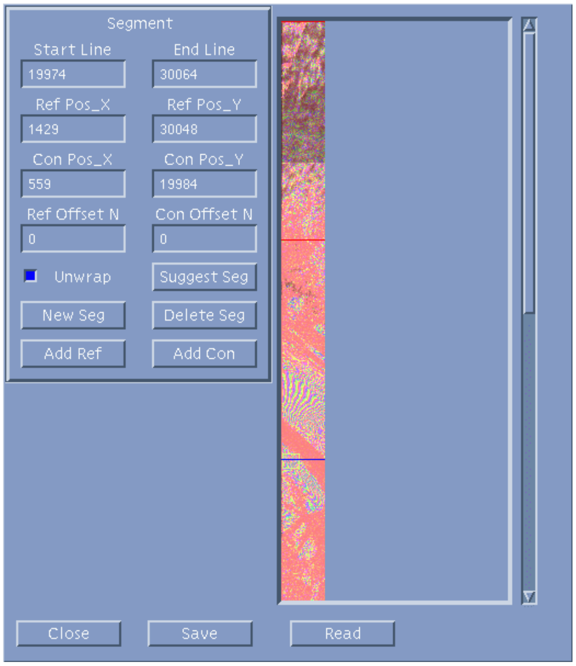
/file/new/Edit Segments
spawns
vView -parm -edit_segment

Parameters: Segment length - integer
default location: VelocityDefaults - segment/ -segment_length
Current Default: 30000
Explanation: to segment the whole swath into smaller blocks in order to ease the memory requirement for create_cuts (Phase unwrapping preparation). This sets the default suggested segment length in the edit segment dialog in vView. Longer segments require longer processing time but less edit time.

This brings up the viewer showing the phase and power as an IHS image.

Bring up the segments edit dialog through the menu Edit/Segments.

If existing segments are to be edited, press the Read button, otherwise get an initial set through the Suggest Seg button which will automatically generate segments with a length determined in the Velocity Defaults by the -segment_length variable in the segment group.



Then you'll have to add the connections between each pair. Click in the first segment and press Add Ref then click near the bottom of the first segment. Place the point in an area where there is a clear fringe in both of the adjacent segments. If the segment bound doesn't cross one of these areas, you can move the segment boundary along the image.

Then click in the next segment and click on Add Con then click in the second segment along the common fringe.

Repeat for each segment pair. Segments can be tagged for not unwrapping by turning off the Unwrap toggle.   This can save processing time in areas such as sea ice.

Save the Segments and exit.

Step 4: New Cuts

This process is an automated process that creates cut lines in areas of low fringes. There are no parameters/options with this step.

/file/new/Cuts

VphaseTool Spawns: rd_velocity -parm

rd_velocity -file $RAMS_HOME/tmp/velocity* -new_mask

create_cuts parm_file

Inspect:
The cut file can be viewed through Gray Scale and select the Cut option.

Step 5: Unwrap Phase

Creates the seeds and unwraps the phase. This is where the BFF file is created.

/file/new/Unwrap Phase

VphaseTool Spawns:

rd_velocity -file $RAMS_HOME/tmp/velocity* -unwrap

doUnwrapping parm_file

Inspect through vView -parm parmfile

You can bring up the Options/Image dialog to work with the histogram and push Apply to scale the image into the local range, or push the Re-Scale to scale it according to the image statistics. You can also bring up the Options/Pixel Info to browse around and inspect the data values.

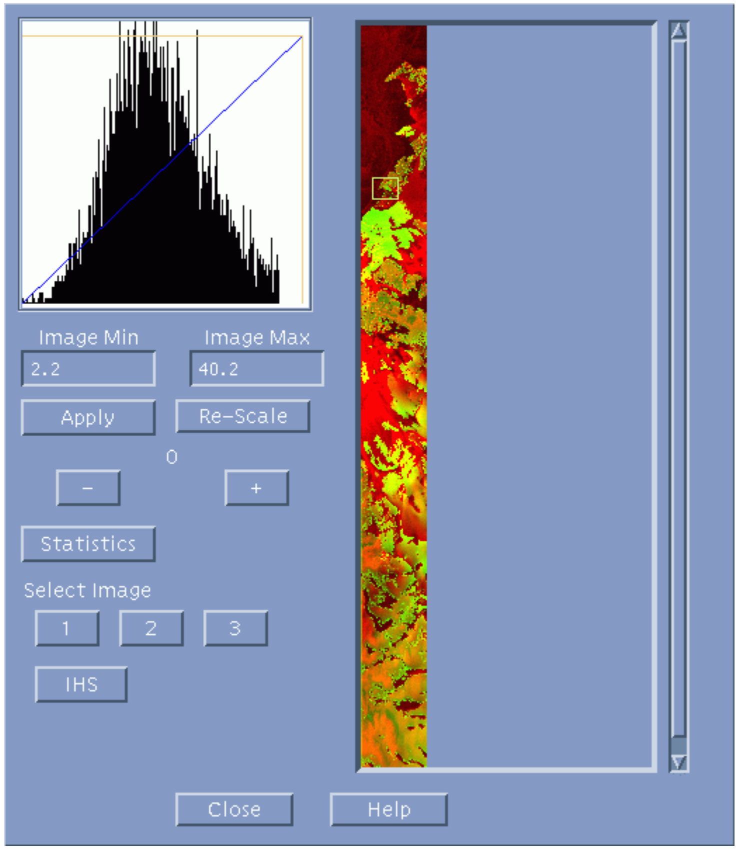
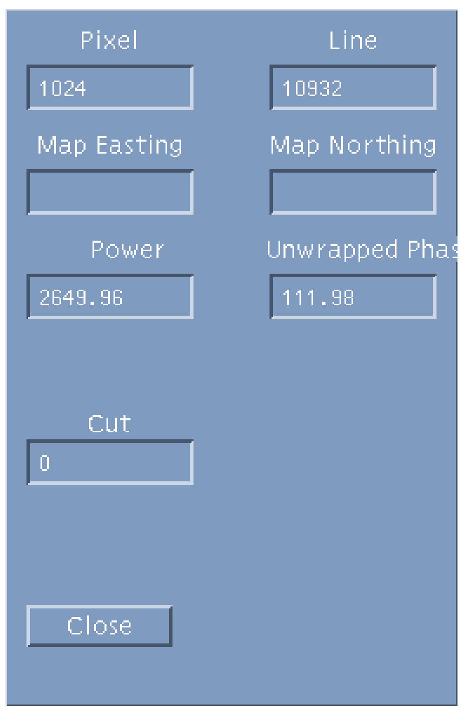


Image Min
Image Max
2.2
40.2
Apply
Re-Scale
0
–
+
Statistics
Select Image
1
2
3
IHS
Close
Help


Pixel
Line
1024
10932
Map Easting
Map Northing
Power
Unwrapped Phas
2649.96
111.98
Cut
0
Close

Parameters:
Variable: unwrapping_mode - enumeration
default location: VelocityDefaults - doUnwrapping / -unwrapping_mode
Current Default: 6
Admissible Values: 2 / 5 / 6
Explanation: generate or update seeds and unwrap the phase image.

Mode index for creating (updating) seeds and doing unwrapping.

Variable: q_mask - float
default location: VelocityDefaults - doUnwrapping / -qmask
Current Default: 0.25
Admissible Values: > 0.0
Explanation: to set all the pixels with lower coherences than the threshold to cuts so that these areas will not be unwrapped. Lower q-mask means more areas unwrapped but with more and bigger unwrapping errors. Coherence threshold to mark the pixels with lower coherence values with cuts. by decreasing the q_mask value (0.1<q_mask<0.25) more areas will be unwrapped. The down side is that more phase unwrapping error.

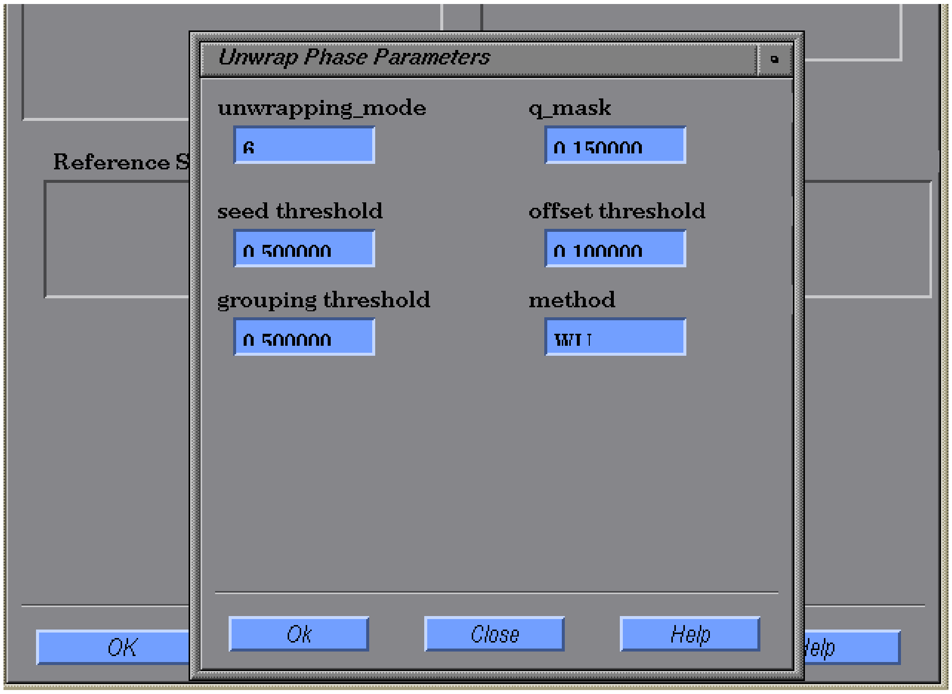
Variable: seed threshold - float
default location: VelocityDefaults - doUnwrapping / -st
Current Default: 0.5
Admissible Values: > 0
Explanation: coherence threshold for selecting the seeds. Only the tie points with higher coherence will be selected as possible seeds.

Variable: offset threshold - float
default location: VelocityDefaults - doUnwrapping / -ct
Current Default: 0.1
Admissible Values: > 0
Explanation: Coherence threshold for offsets (tie points with lower coherence will not participate seed generation)

Variable: grouping threshold - float
default location: VelocityDefaults - doUnwrapping / -gt
Current Default: 0.5
Admissible Values: > 0
Explanation: used for deciding which phase island will the tie point belong to. If more pixels than the percentage threshold (0.5 means 50%) of pixels within the area of the tie point chip belongs to one phase island, this tie point will be grouped to this phase island. Otherwise this tie point will not belong to any islands.

Coherence threshold for tie point grouping.

Variable: method - string
default location: VelocityDefaults - doUnwrapping / -method
Current Default: WU
Admissible Values: WU
Explanation: There is only one method available now.



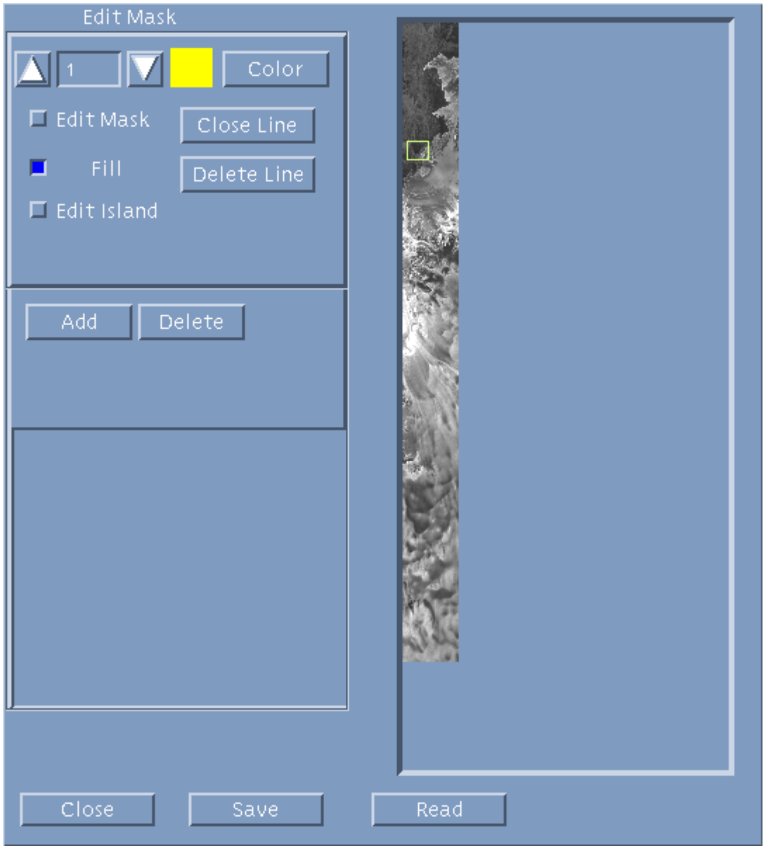
The BFF file includes cut info and unwrapped phase info. Viewing the BFF file with PHASE a visual inspection can be made for areas that have the potential to be unwrapped (fringes) but were not. A decision is then made as to whether to decrease the q_mask so that these areas get unwrapped.

Step 6: Tide Model **used for RAMP processing**

Creates the tidal mask & tide points. There are no options/parameters with this step.

vView -edit_tide

To start defining a mask, push Edit Mask. Start tracing the mask but don't let the line cross itself. You can scroll around if the sea ice is bigger than the current window. Finish by pressing close. "Fill" performs a flood fill in with the current color for the masked area. Once the mask is defined, add a number of points through pushing Add then selecting a point on the screen.



Step 7: Create VGCPs

VphaseTool Handle:

/file/new/Add Vcp

VphaseTool Spawns:

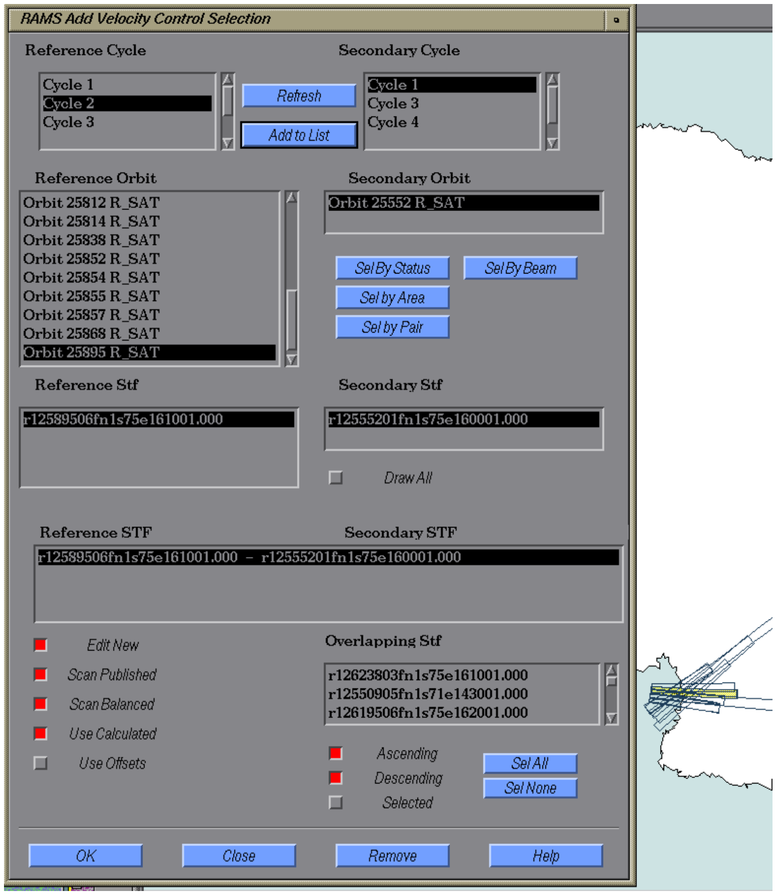
rd_velocity can be used to do an automatic search among the existing published vcps or if that fails to produce vcps for a given swath, the editor can be used to designate a know 0 velocity point as a vcp. Otherwise one or more velocity points will need to be chosen from overlapping pairs already calculated. There has to be a velocity control point before processing can continue. If none of the existing vcps fall within a frame, use the editor.

rd_velocity -file $RAMS_HOME/tmp/velocity* -add_vcps
or
vView -file parm_file -edit_vcp

Wrapper program spawns:

generate_vgcp dem_data dem_par interferogram.par global_vcp_input vcp_output

This gathers all VGCPs that are contained within the boundaries of a swath. Any combination of VGCP sources can be selected.



Balanced Velocities-	ON
Published Velocities-	ON (ERIM, NSIDC, Zero VCPs)
Calculated- velocities.	ON Tie points obtained from overlapping swaths with calculated
	Options are to choose which overlapping swaths to include.
Use Offsets-	OFF.
Edit New-	OFF This option is used when the operator wants to manually add VCPs.

Procedure:

Variable: Use Offsets - boolean
default location: VelocityDefaults - generate_vgcp / -uo
Current Default: 0

Admissible Values: 0 / 1

Explanation: To use (1) or not (0) the range offsets wherever the phase is not unwrappable for refine baseline in the case velocity map products are used to create VCPs.

Step 8: Estimate Baseline

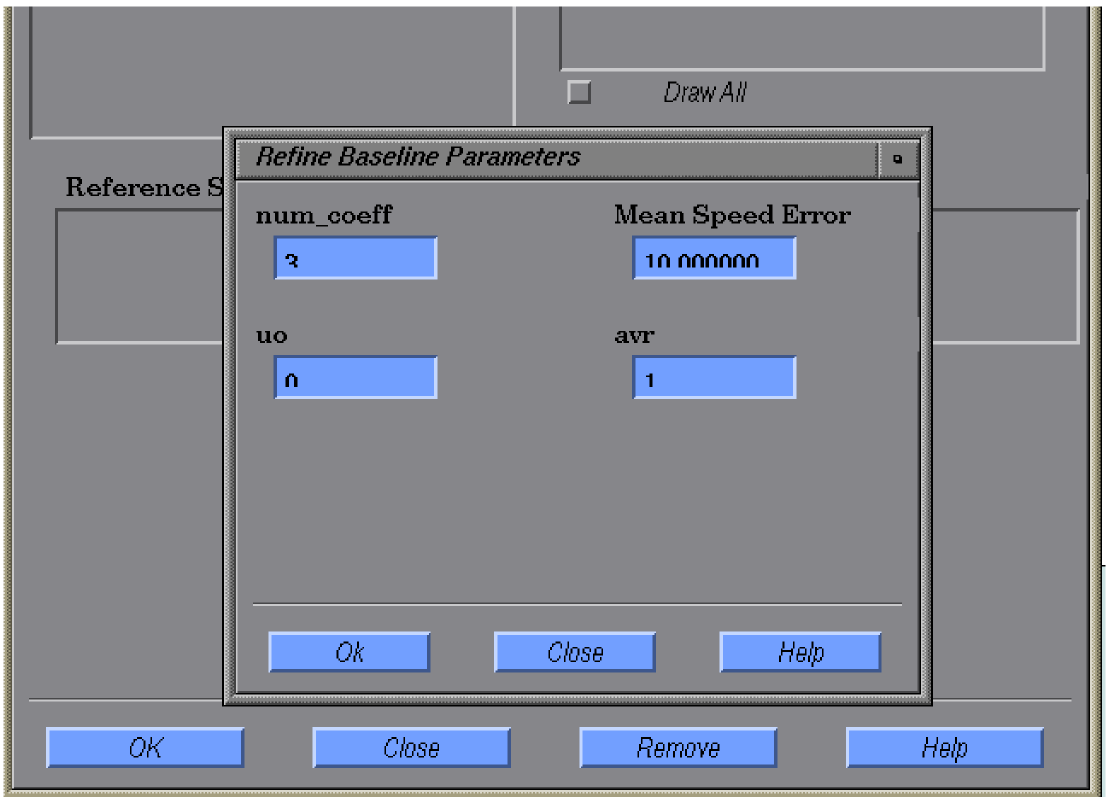
This refines the baseline based on the velocity control points.

VphaseTool Spawns:

rd_velocity -parm -baseline

Wrapper program spawns:

removeDEM -parm parm_file



Parameters:

Variable: num_coeff - integer
default location: VelocityDefaults - removeDEM / -num_coeff
Current Default: 3
Admissible Values: 1 ~ 4

Explanation: There are altogether 6 unknown parameters. But baseline refinement we only refine 4 (parallel, perpendicular, parallel rate, perpendicular rate) of them with the two others (along track, along track rate) unchanged.

Variable: Mean Speed Error - float
default location: VelocityDefaults - removeDEM / -mse
Current Default: 5.0
Admissible Values: > 0
Explanation: This is the limit of average speed error. If the average speed error for the used VCPs is bigger than this limit, it will not continuing the process (if avr = 0) or will continue by delete the VCP with biggest error until the error is within the limit (if avr = 1).

Variable: uo - boolean
default location: VelocityDefaults - removeDEM / -uo
Current Default: 0
Admissible Values: 0 / 1
Explanation: To use (1) or not (0) range offsets to replace unwrapped phase wherever phase is not unwrappable.

Variable: avr - Boolean
default location: VelocityDefaults - removeDEM / -avr
Current Default: 1
Admissible Values: 0 / 1
Explanation: avr means automatic VCP removal. If avr = 0, manual mode; If avr = 1, automatic mode.

Procedure (?)

Start by using -all- the available VGCPs (avr = 0, uo=1) ???
Options:
- Use balance velocities only near divides
- Edit/move Zero Velocity VGCPs to optimize location
- Eliminate VGCPs with high errors.

After each edit, re-run estimate baseline to reassess errors and VGCP distribution.

Criteria:
- Good overall distribution.
- Good overall mean speed error

Step 9: Remove DEM

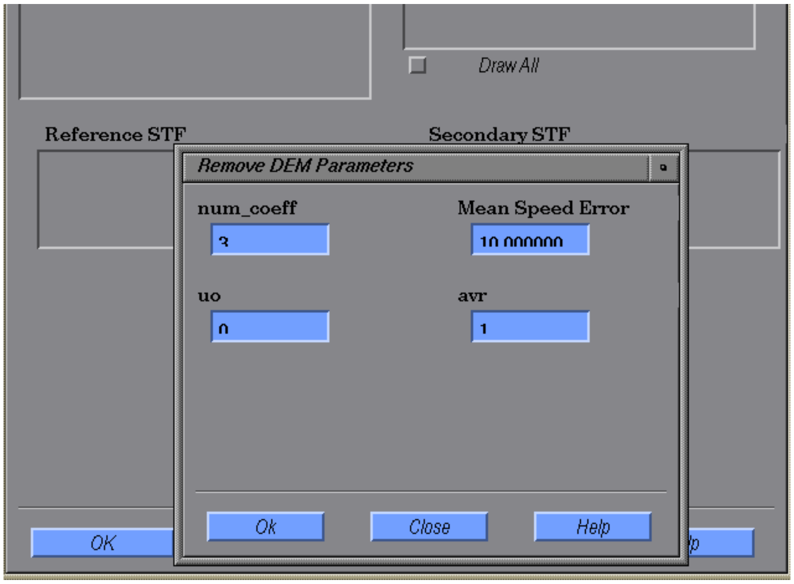
/new/file/Remove DEM

VphaseTool Spawns:

rd_velocity -parm -remove_dem

Wrapper program spawns:

removeDEM -parm parm_file



The variables for the Remove Dem step are the same as for Refine Baseline.

Procedure:

We want to use the OFFSETS (uo=1) when removing DEM to get estimates of absolute velocity in regions where we only have speckle retracking estimates of the displacements.

Step 10: Estimate Velocity

VphaseTool Handle:

/new/file/Estimate Velocity

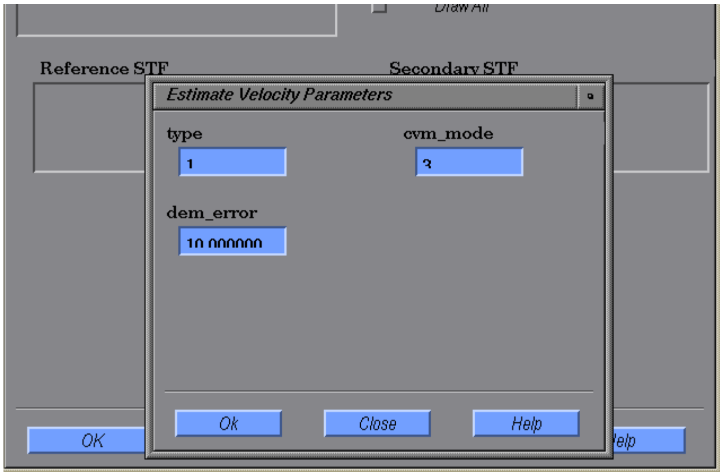
VphaseTool Spawns:

rd_velocity -parm parm_file -velocity

Wrapper program spawns:

create_velocity_map parm
generate_cvm parm to create a tiff file of the velocities.

The tiff file can also be generated explicitly through File/New/Velocity Tiff



type- 1
cvm_mode- 3
dem_error 10.000

Parameters:

Variable: type - boolean (it is now ignored. Both types of products will be created at the same time)
default location: VelocityDefaults - create_velocity_map / -type
Current Default: 1
Admissible Values: 0, 1
Explanation: type of velocity map product (0: use offsetmap only; 1: use azimuth offset and phase)

Variable: cvm_mode - enumeration
default location: VelocityDefaults - create_velocity_map / -cvm_mode
Current Default: 3
Admissible Values: 0 ~ 3
Explanation: vertical movement mode
0: assuming zero vertical movement;
1: assuming vertical movement coming from the local slope;
2: assuming vertical movement coming from the tide;
3: assuming vertical movement coming from the tide plus local slope;

Variable: dem_error –float

default location: VelocityDefaults - create_velocity_map / -dem_error
Current Default: 10.0
Admissible Values: >= 0
Explanation: error of the DEM used in meters

Parameters for generate_cvm

Variable: -ex_scale float
default location: VelocityDefaults - generate_cvm /-ex_scale
Current Default: 0.35
Admissible Values: > 0
Explanation: exponential scale. Output = input^{ex_scale}

Variable: -vector_scale float
default location: VelocityDefaults - generate_cvm /-vector_scale
Current Default: 0.1
Admissible Values:
Explanation: velocity * vector_scale = pixels occupied

Variable: -arrow_value integer
default location: VelocityDefaults - generate_cvm /-arrow_value
Current Default: 0
Admissible Values: 0 ~ 255
Explanation: This value will be used to draw the arrows (0 means black and 255 means white).

Variable: -dec_x integer
default location: VelocityDefaults - generate_cvm /-dec_x
Current Default: 1
Admissible Values: > 0
Explanation: decimation factor in x direction (easting) deciding how dense the vectors will be drawn on the background. 1: every tie point will be drawn; 2: one vector will be drawn for every two tie points; and so on.

Variable: -dec_y integer
default location: VelocityDefaults - generate_cvm /-dec_y
Current Default: 1
Admissible Values: > 0
Explanation: decimation factor in y direction (northing) deciding how dense the vectors will be drawn on the background. 1: every tie point will be drawn; 2: one vector will be drawn for every two tie points; and so on.

Variable: -no_data_replacement float
default location: VelocityDefaults - generate_cvm / -no_data_replacement
Current Default: 0.0
Admissible Values: 0 ~ 255

Explanation: This value will be used wherever no data is available.

Variable: -indicator Boolean
default location: VelocityDefaults - generate_cvm / -indicator
Current Default: : 0
Admissible Values: 0 / 1
Explanation: 0: horizontal bar indicator; 1: vertical bar indicator

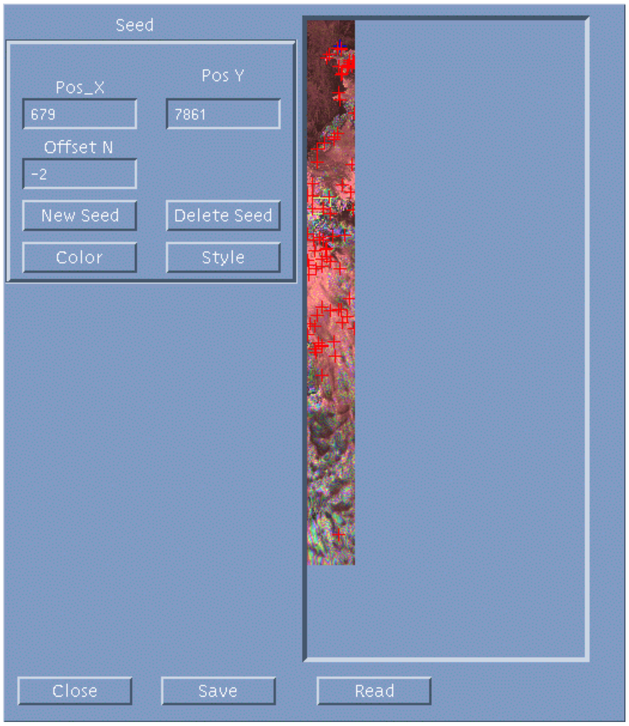
Step 11: Review Phase Island Reconciliation

If velocities are not being calculated in regions of good fringes separated from VCPs by decorrelation bands:

a) Display BFF and PHASE (Intensity and Hue) and overlay with velocity vectors.
b) Add seeds where phase is not being unwrapped. Count fringes to determine seed values.

The seed editor is called through Edit/Seeds

It will read seed file for pair if available and allow interactive editing. If the seed file is not available it will allow creation of new one.



Repeat Phase unwrapping
Repeat Est. Baseline ???
Repeat Remove DEM
Repeat Estimate Velocity

Step : Velocity Tiff

Parameters: No Options

Step : Mosaic Velocity

Parameters: Like the "Calculated VCP dialog, the user can hand select which overlapping datatakes will be used to merge velocities.

Step 12: Mosaic Velocities

/New/file/Mosaic Velocity

Velocity mosaics are stored in the database under ifr/overviews/ov_*

The current interface allows the user to specify an area through a box onthe screen and specify a name. This is undergoing some revision to allow the user to hand select datatakes contributing to the mosaic.

Before Conflation (merge ascending/descending) 4 velocity maps are created for each STF pair:

Regular (phase/azimuth offset)
Offset-only
Ascending-descending azimuth offset only
*Ascending-descending range only

*The ascending-descending range velocity map is created using the 4 possible combinations of:

phase/phase
phase/offset
offset/phase
offset/offset in that sequence

Conflation uses the 4 possible products from each STF pair.

If the option ‘-use_azimuth_offset’ is set to 0 (default) the azimuth only prodeuct will NOT be used in the conflation.

If the option ‘-use_azimuth_offset’ is set to 1, all 4 of the products will be used.